Oracle FLEXCUBE Direct Banking

Development Workbench for Direct and Mobile Banking User Manual
Release 12.0.3.0.0

Part No. E52543-01

April 2014

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway
Goregaon (East)
Mumbai, Maharashtra 400 063
India
Worldwide Inquiries:
Phone: +91 22 6718 3000
Fax: +91 22 6718 3001
www.oracle.com/financialservices/

Contents

Preface

Intended Audience

Oracle FLEXCUBE Direct Banking Development Workbench document in particular is targeted towards the following groups of users.

- Oracle FLEXCUBE Direct Banking Development Teams
- Oracle FLEXCUBE Direct Banking Implementation Teams
- Oracle FLEXCUBE Direct Banking Implementation Partners

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc

Access to OFSS Support

https://support.us.oracle.com

Structure

This manual is organized into the following categories:

- *Preface* gives information on the intended audience. It also describes the overall structure of the User Manual
- *Overview* provides brief description of the tool.
- *Chapters post overview* explains different functionality of the tool and steps to use them.

Related Information Sources

For more information on Oracle FLEXCUBE Direct Banking Release 12.0.3.0.0, refer to the following documents:

- Oracle FLEXCUBE Direct Banking Licensing Guide.
- Oracle FLEXCUBE Direct Banking User Manuals.

Overview

Oracle FLEXCUBE Development WorkBench for Direct and Mobile Banking is an Integrated Development Environment (IDE) for Oracle FLEXCUBE Direct Banking (FCDB) Development. This IDE provides the platform to the user to do following operations

- ✓ Writing FCDB Business Services using the predefined FCDB Java Project Template in the IDE.
- ✓ Designing FCDB Screens.
- ✓ CSS and Image Editing.
- ✓ Transaction and Service Registration into FCDB Database Tables.
- ✓ Wizards for some of the configuration tables related to transaction development in FCDB. User can Add/Modify/Delete these properties directly from the IDE.

Prerequisites

Following are the prerequisites to use this tool

1. To use this tool one should have “JDK 7” or higher version of java on their machine.

2. One of the following files to install the tool.

   a) **channelworkbench-windows.exe**
      An installation file for Windows Operating System.

   b) **channelworkbench-linux.sh**
      An installation file for Linux Operating System.

   c) **channelworkbench.zip**
      A compressed zip file containing Workbench tool files to be extracted to target folder and run.

For more details on installation procedure of tool, please refer to:

Oracle FLEXCUBE Development Workbench for Direct and Mobile Banking Installation Manual

How to run the tool

This tool can be run in following ways:

- To run the tool double click the following icon on Desktop,



OR

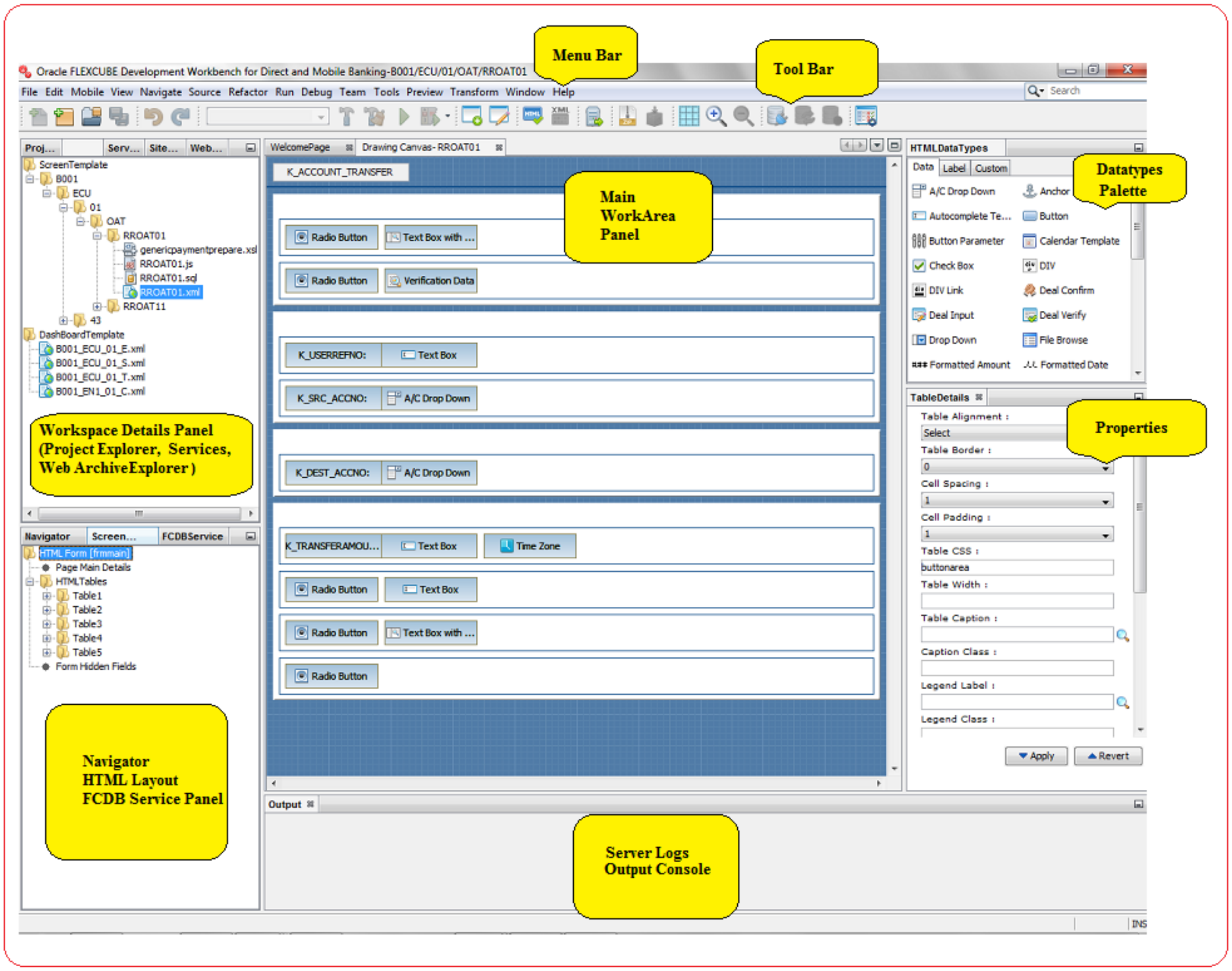
- Double click on **“channelworkbench.exe”** file in the bin folder of the tool.

The user can also search for tool in Start Menu by keyword **“Development Workbench”** and select the tool to launch.

The tool will be launched as shown below:



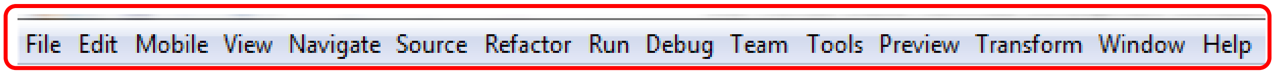
Once the progress bar completes, splash screen goes off the screen and the main working platform appears. The following screenshot displays the main working platform along with the description of the different panels.



In the following section, details of each of the highlighted components of this tool are discussed.

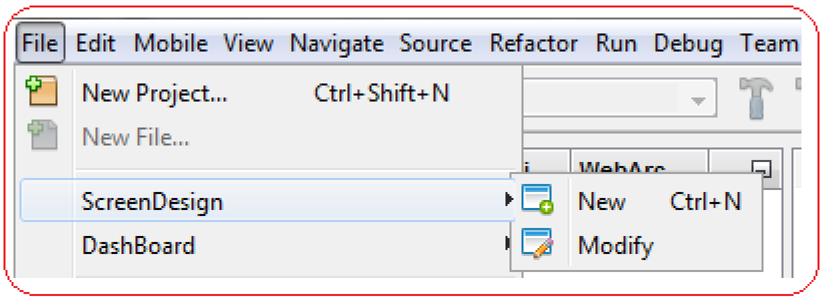
Menu Bar

The Menu Bar of Development Workbench tool contains default Netbeans IDE Menu items as well as few Menu items specific to the tool.

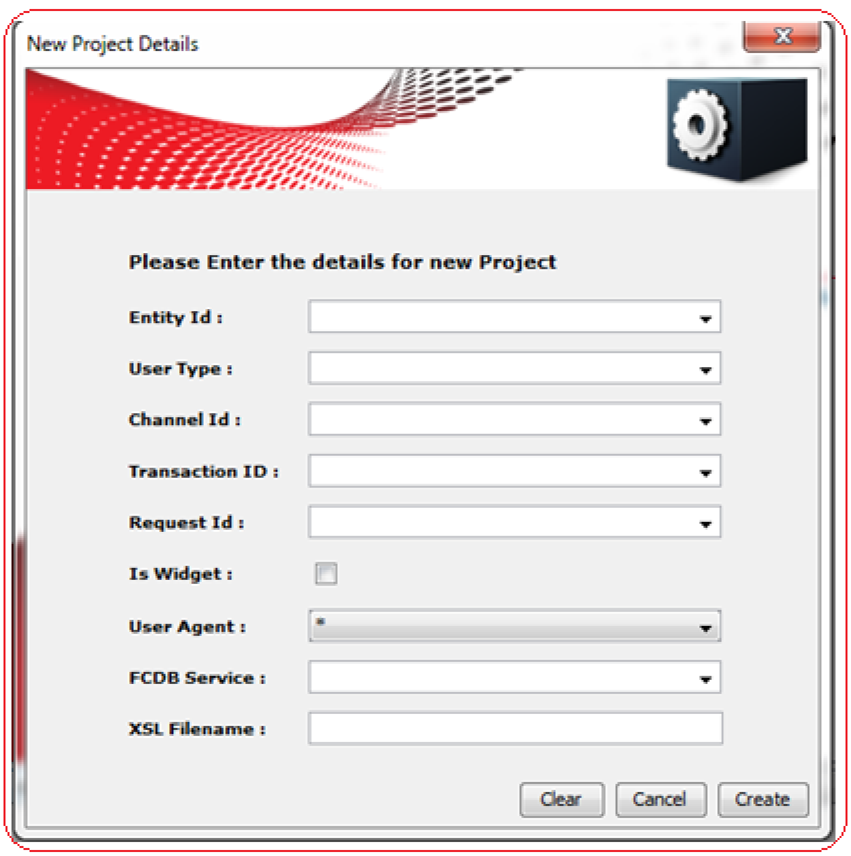


The list of all Menu items specific to the tool is described below:

1. **<u>File → Screen Design:</u>** Screen Design menu item in File Menu has following two sub menus

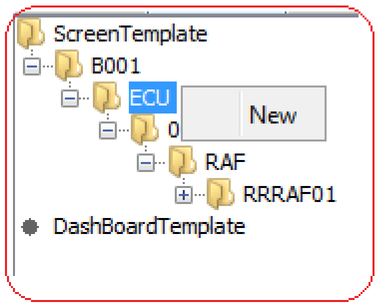


   a. **<u>New:</u>** This sub-menu allows a user to create a new Screen Template project in the workspace of the tool. Clicking on "New" opens a dialog window as shown below

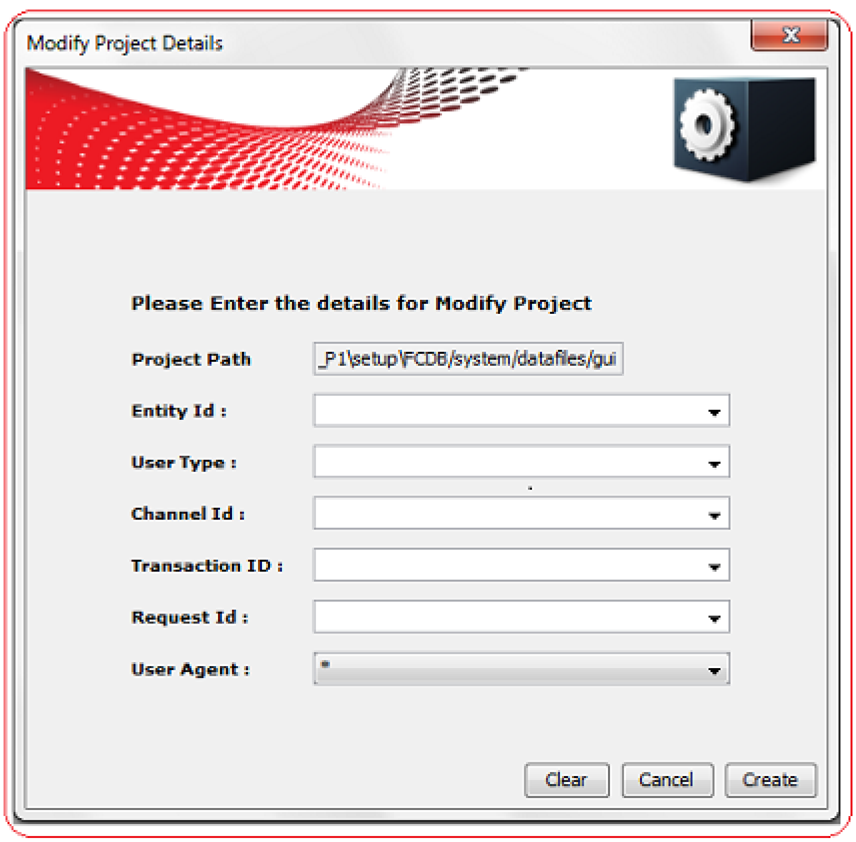


User needs to enter the entity id, usertype, channel id, transaction id, request id, user agent, fcdb service and XSL file name of the screen to be designed. Click on “Create” to create a new project in the workspace. Click on “Cancel” to close this dialog. Click on “Clear” to clear the input fields.

A new Screen Template project can also be created by right clicking on the file tree node and selecting “New” option as shown below:

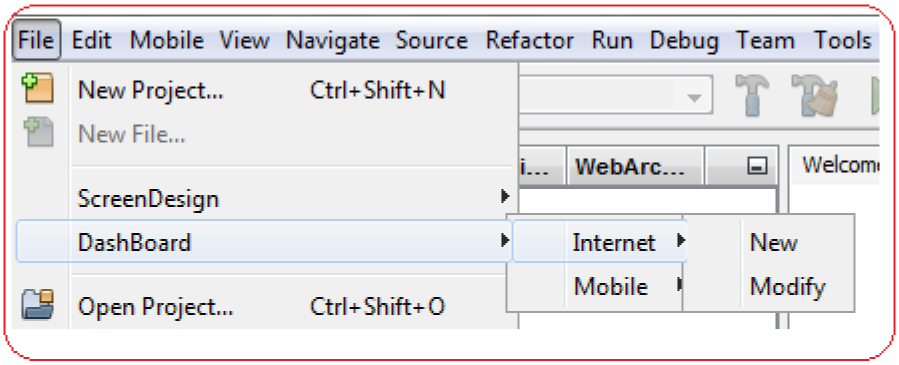


b. **Modify:** This sub-menu allows a user to modify the existing screens in FCDB application. Using this sub-menu user can load the Screen Template xml from FCDB datafiles folder into the workspace of this tool and modify the screen design. Clicking on “Modify” opens a dialog window as shown below.



Project Path consists of the default FCDB home folder path where gui/services are kept. User needs to enter the entity id, usertype, channel id, transaction id, request id, user agent and FCDB service name of the screen to be modified. Click on "Create", on the basis of the input field's tool will copy the screen template xml and js file from FCDB project into its own workspace. Click on "Cancel" to close this dialog. Click on "Clear" to clear the input fields.

2. **File → Dashboard:** This sub-menu allows user to create/modify Dashboard Projects.



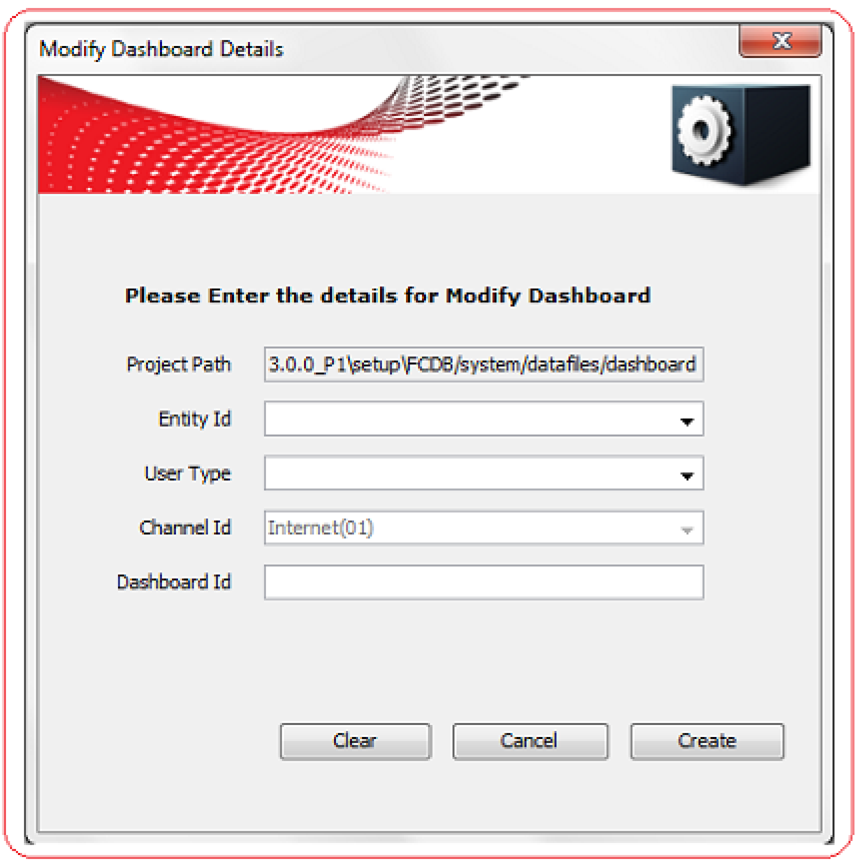
I. Internet

a. **New:** Clicking this sub-menu, user can create a new Dashboard Project for Internet channel.



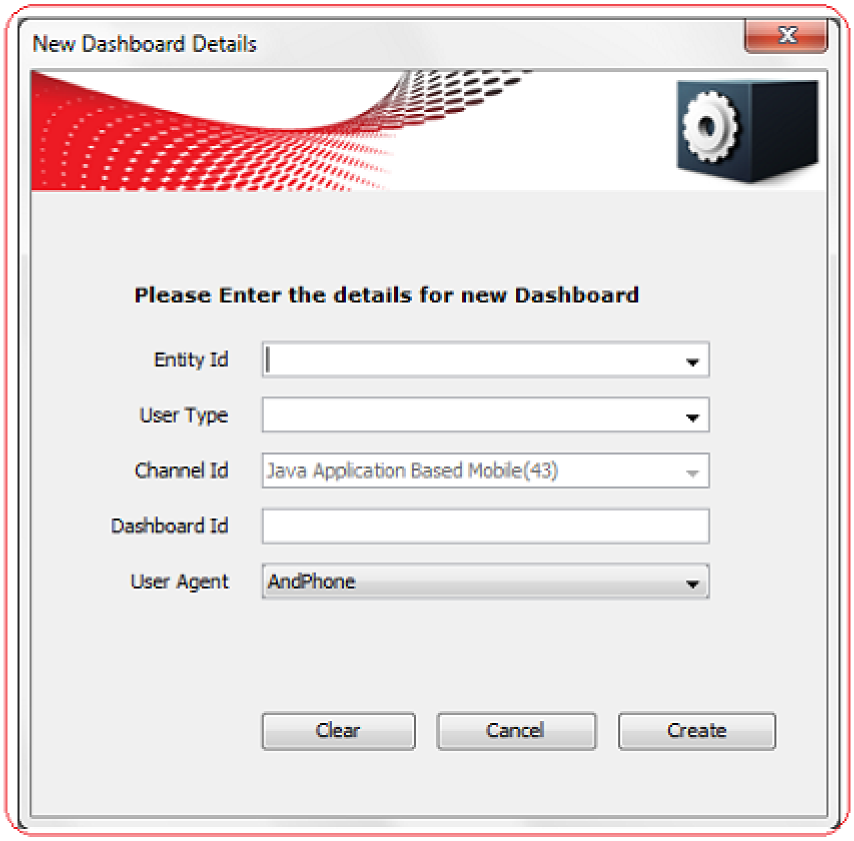
- ✓ **Entity id:** Enter the IdEntity for which the screen is being created. e.g. "B001".
- ✓ **User Type**: Enter the UserType for which the screen is being created. e.g. "ECU".
- ✓ **Channel Id**: Channel ID is a non-editable field i.e. it is auto populated with Internet Channel ID "01" since DashBoard design for Internet Channel was selected.
- ✓ **Dashboard ID:** Dashboard ID for which dashboard is to be created.
- ✓ **Is Simple Dashboard:** This field represents a flag to indicate if the dashboard is simple or not. Simple Dashboard supports only top positioned widgets. If simple dashboard is true, then Landing request ID is mandatory. This field will not be considered in create and modify user.
- ✓ **Is PFM Dashboard:** This field represents a flag to indicate if the dashboard is designed for PFM Transaction or not.

b. Modify: Clicking this sub-menu, user can modify an existing Dashboard Project for Internet channel.



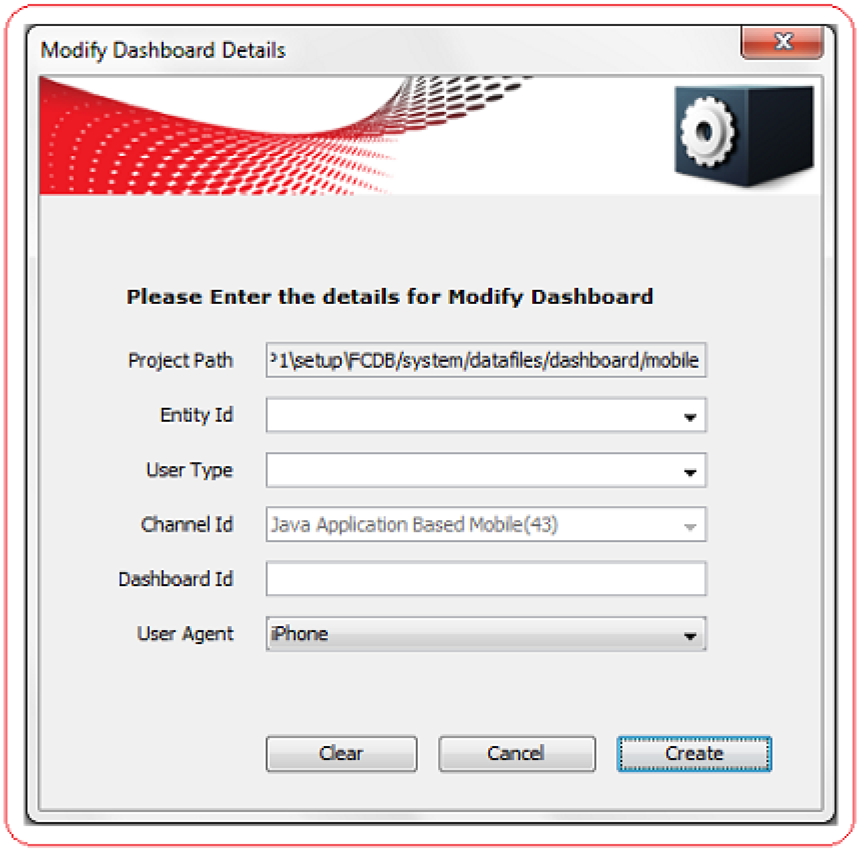
II. Mobile

a. New: Clicking this sub-menu, user can create a new Dashboard Project for Mobile channel**.**

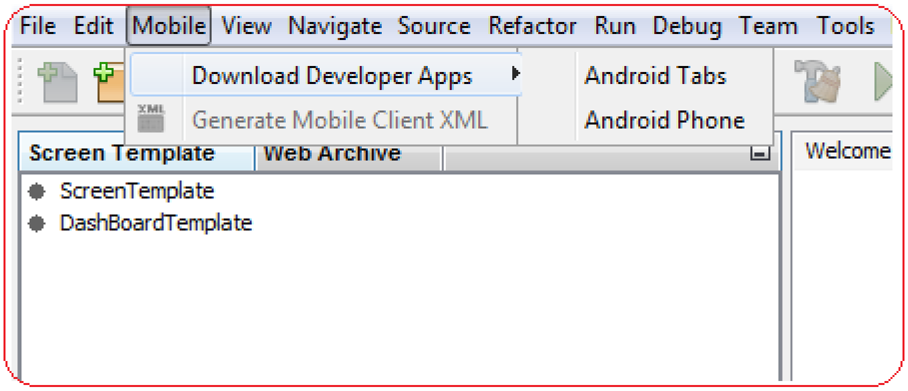


- ✓ **Entity id:** Enter the IdEntity for which the screen is being created. e.g. "B001".
- ✓ **User Type:** Enter the UserType for which the screen is being created. e.g. "ECU".
- ✓ **Channel Id:** Channel ID is a non-editable field i.e. it is auto populated with Mobile Channel ID "43" since DashBoard design for Mobile Channel was selected.
- ✓ **Dashboard ID:** Enter the Dashboard ID for which the screen is being created. e.g. "RRPFM61".
- ✓ **User Agent:** Select the User Agent from the drop down for which the screen is being created. e.g. "AndPhone".

b. **Modify:** Clicking this sub-menu, user can modify an existing Dashboard Project for Mobile channel.



3. **Mobile:** This Menu Bar item is specific to Mobile Application Channel (Channel 43).



It consists of the following two options:

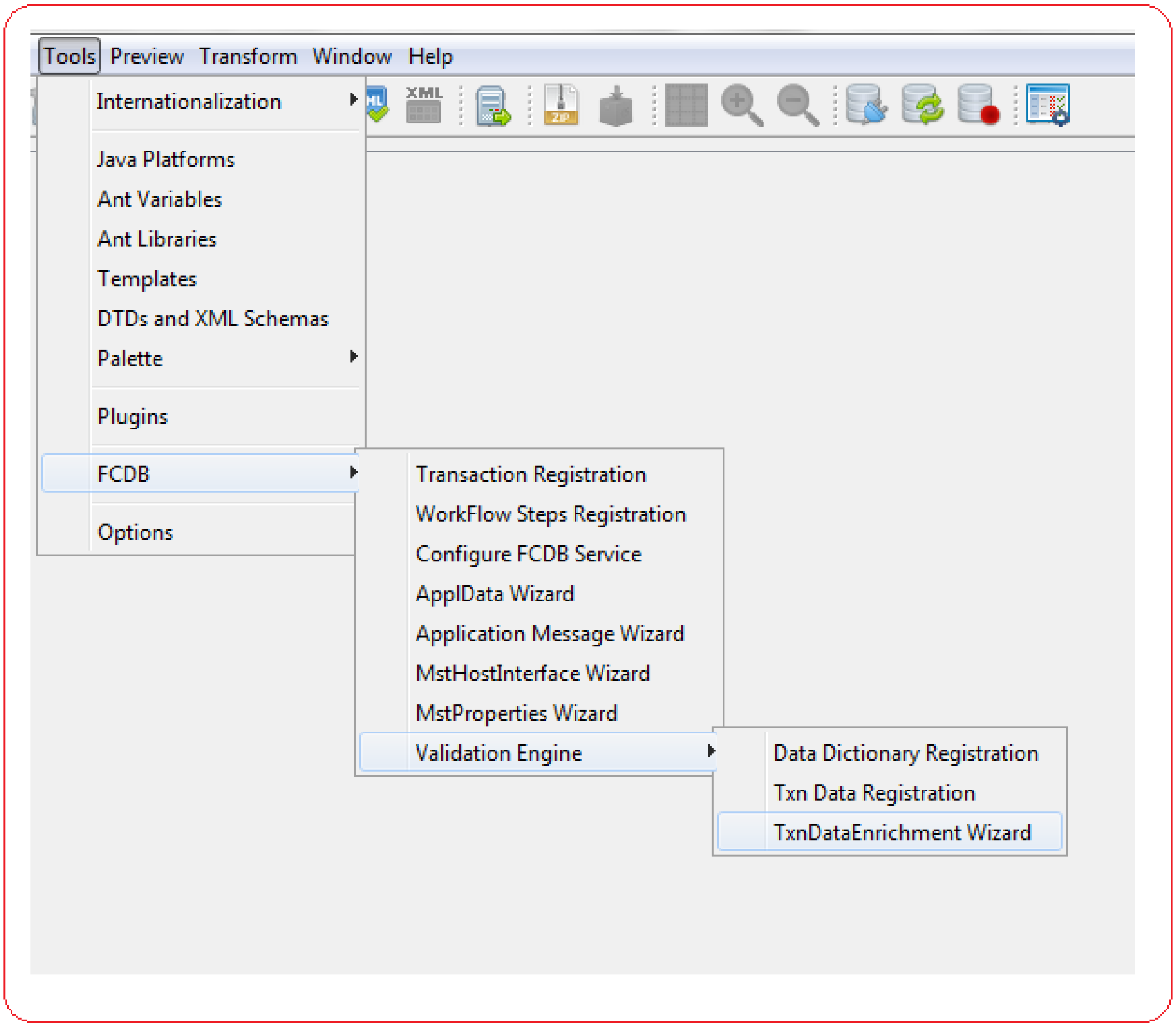
a. **Download Developer Apps:** Clicking on the submenu opens another submenu containing application of various user agents.

   a. **Android Tabs:** Clicking this submenu downloads the FCDB application (.apk file) supported by Android Tablets.

   b. **Android Phone:** Clicking this submenu downloads the FCDB application (.apk file) supported by Android Phone.

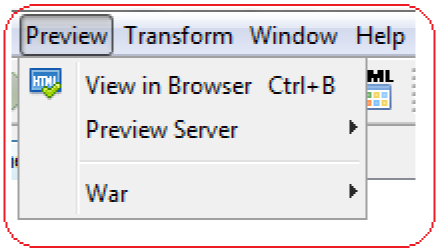
b. **Generate Mobile Client XML:** Clicking this sub- menu generates the mobile client XML.

4. **Tools → FCDB:** This sub menu lists all available FCDB wizards for some of the configuration tables related to transaction development in FCDB.



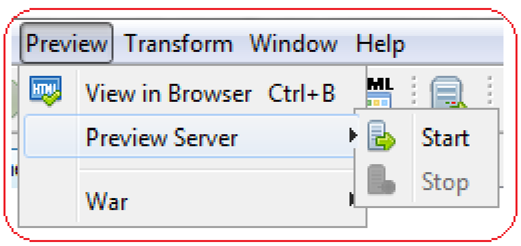
More detailed description of each wizard is given in "**FCDB Application Configurations**" section of the document.

5. **Preview:** Preview menu has following sub-menus.

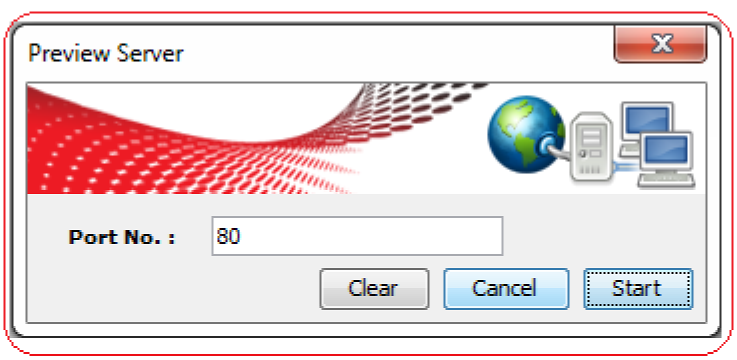


a. **View in Default Browser:** Clicking this sub-menu will launch the system's default browser with the output html of the current screen design project. User can preview the screen design done by them in this output html.

b. **Preview Server:** This menu allows the user to start/stop server of tool.

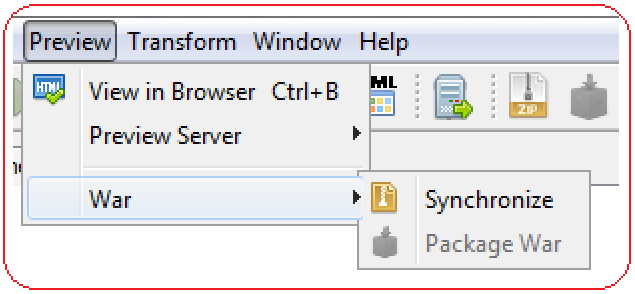


On clicking start the following dialog pops up,



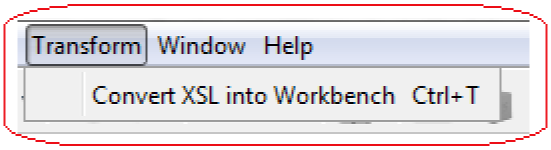
Enter the port number of the server. Click on "Start" to start the server of tool. Click "Cancel" to close this dialog. Click on "Clear" to clear port no.

c. **War:** This menu allows user to synchronize / package War.

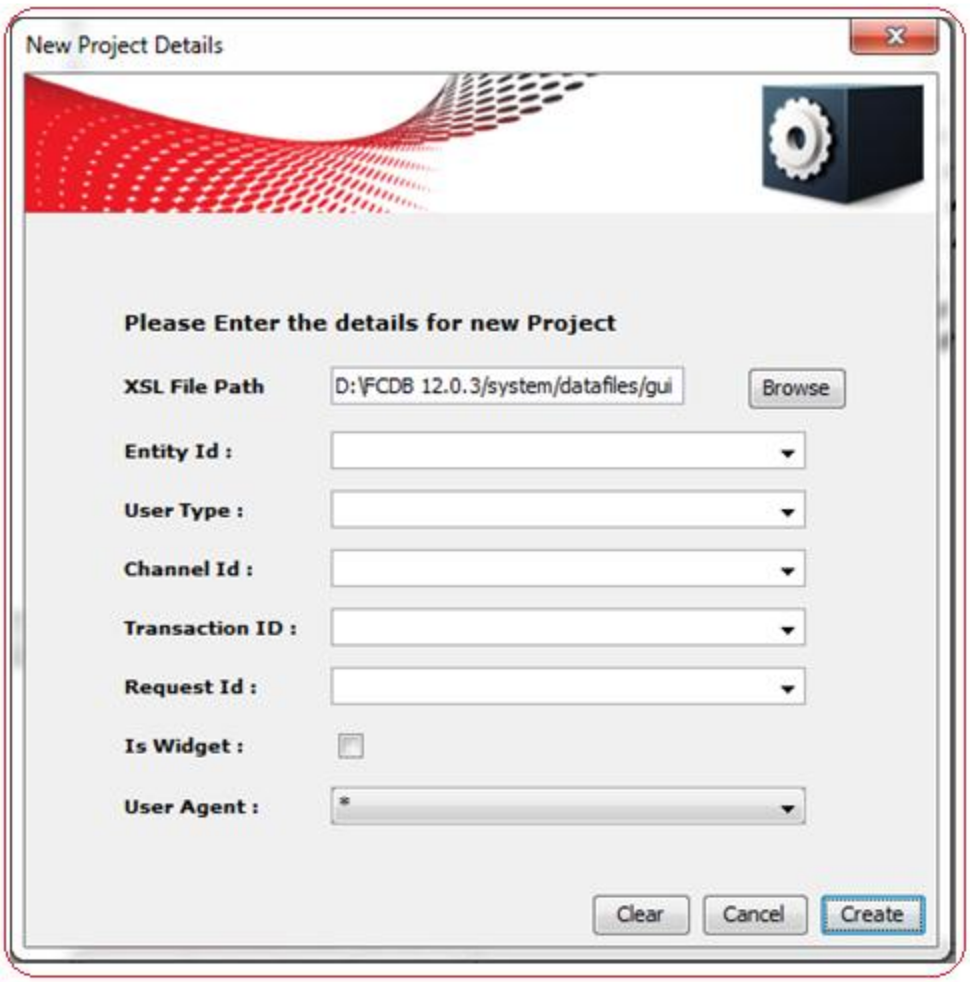


More detailed description of Synchronize/Package war is given in the "**Synchronize/Package WAR**" section of the document.

6. **Transform:** Transform menu has following sub-menu



a. **Convert Xsl into Workbench:** This sub-menu allows a user to convert existing FCDB screen which has been designed using XSL file to Channel Workbench framework. Clicking this sub-menu opens a dialog window as shown below.



User needs to browse the XSL file path, enter the entity id, usertype, channel id, transaction id, request id and user agent of the screen to be converted to Channel Workbench framework. Click on “Create”, on the basis of the input field’s tool will convert the XSL file into Screen Template xml and js file and load these files into its own workspace. Click on “Cancel” to close this dialog. Click on “Clear” to clear the input fields.

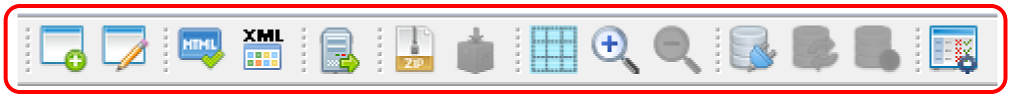
While transforming XSL to Channel Workbench, tool will generate some self-explanatory messages in the output log file available in the directory logs as channelworkbenchout.log.

7. **Help → About:** Clicking this sub-menu opens following dialog window which contains the details associated with the tool version and working directories.



Tool Bar

Tool Bar of this tool has the following icons.



a.  New: This icon allows a user to create a new screentemplate project in the workspace of the tool.

b.  Modify**:** This icon allows a user to modify the existing screens in FCDB application. Using this icon user can load the screentemplate xml from FCDB datafiles folder into the workspace of this tool and modify the screen design.

c.  Preview**:** This icon allow user to preview the complete screen layout at any point of change done by them. Previewing launches the browser with the output html. One can preview either in system default browser or select the alternative browser .exe path and launch the alternative browser. Previewing automatically saves your screen definition.

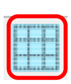
d. 

Generate Mobile Client XML**:** This icon allows user to generate mobile client XML for channel id 43.

e.  Server Start/Stop: This icon allows the user to start/stop server of tool.

f.  Synchronize war : This icon allows user to synchronize latest war in to the workplace.

g.  Package war**:** This icon allows user to repackage the war with updated CSS, JS and image files. Package War will be enabled only if any Web component like CSS, JS or images files has been edited by the user.

h.  Show Canvas: This icon shows Drawing Canvas for user Agents of Channel 43 or Wireframe Canvas for channels other than Mobile Channel.

i.  Zoom in**:** This icon allows user to zoom into the canvas

j.  Zoom Out: This icon allows user zoom out of the canvas.

k. Database Connectivity:

i.  Database Connection: This Icon signifies that you are connected to Database. Click this icon to modify database connection.

ii.  Database Refresh: Click this icon to refresh the database connection and reload the changes done in database.

iii.  Disconnect Database Connection: Click this icon to close the database connection.

l.  FCDB Setup Details: FCDB Properties dialog will allow user to enter the path of the FCDB folder and the language.

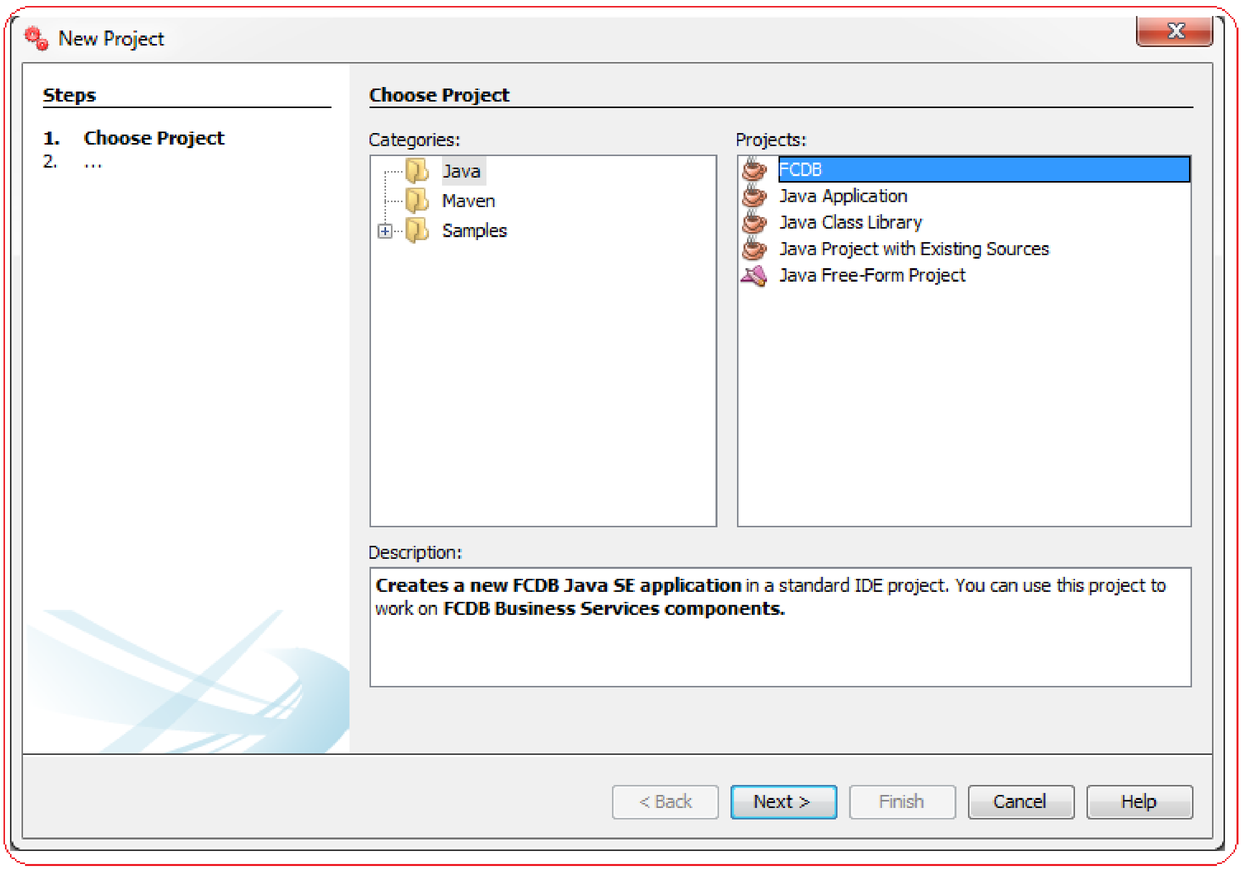
Workspace Details Panel

Workspace Details Panel is displayed on the top left corner of the tool. The list of available windows in this panel is described below:

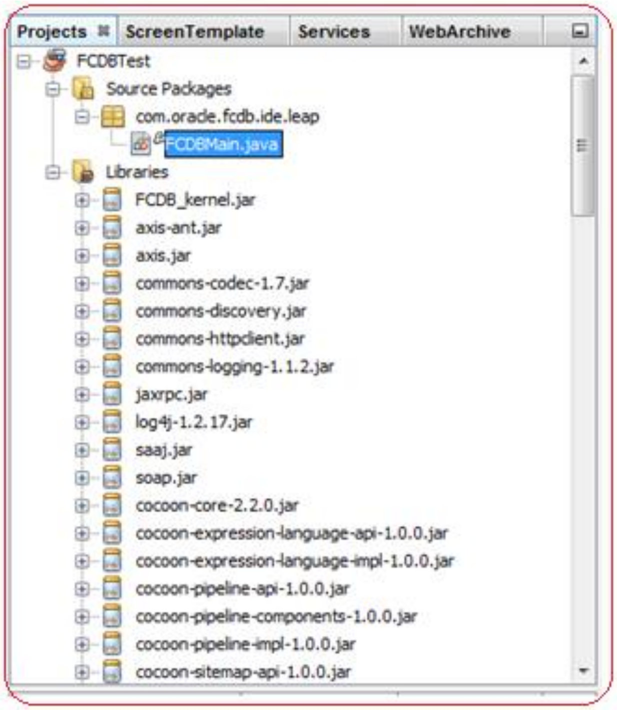
1. Projects Window

Project Window allows user to create new Java /FCDB project or open an existing Java/FCDB project in the tool. The tool provides all the necessary features to build, run and debug the project.

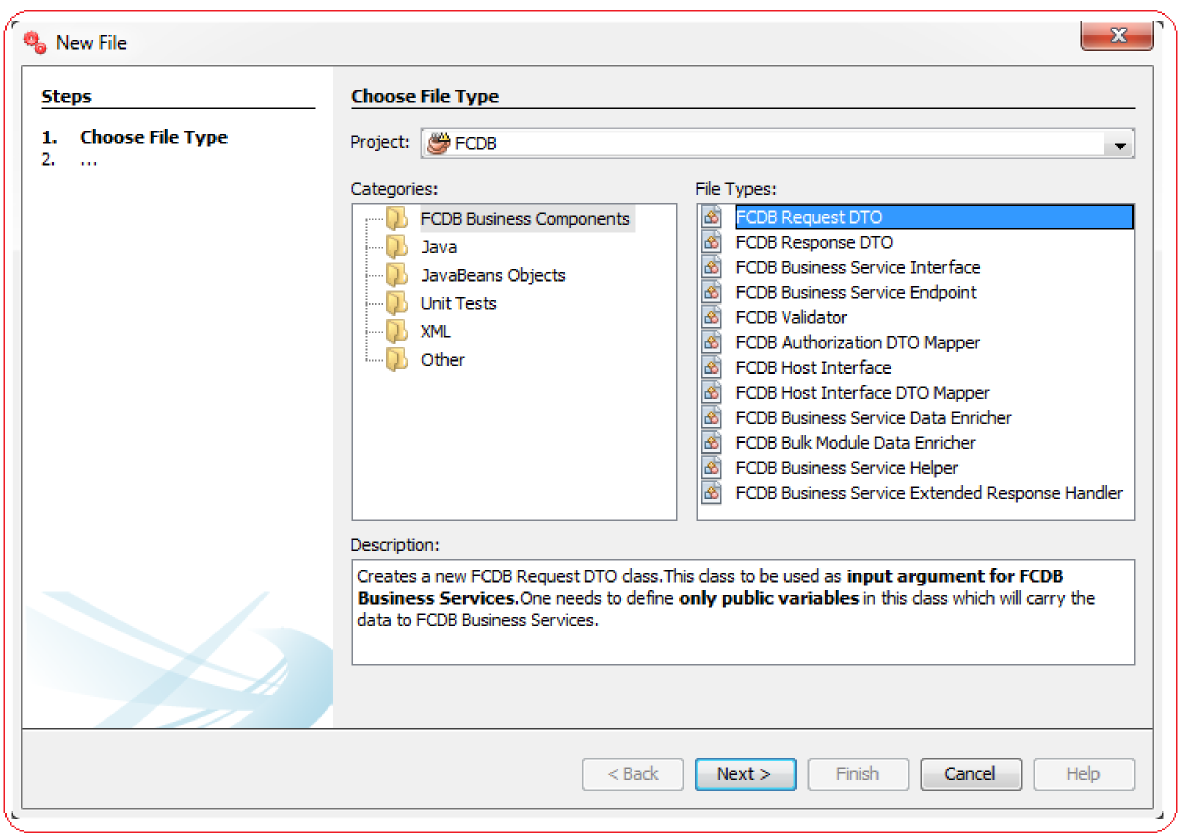
FCDB project option creates a new FCDB Java SE application in a standard IDE project. User can use this project to work on FCDB Business Services components.



On creating a new FCDB Project, all the necessary jars required to work on FCDB Business Services components are loaded in the libraries of the project as shown below:

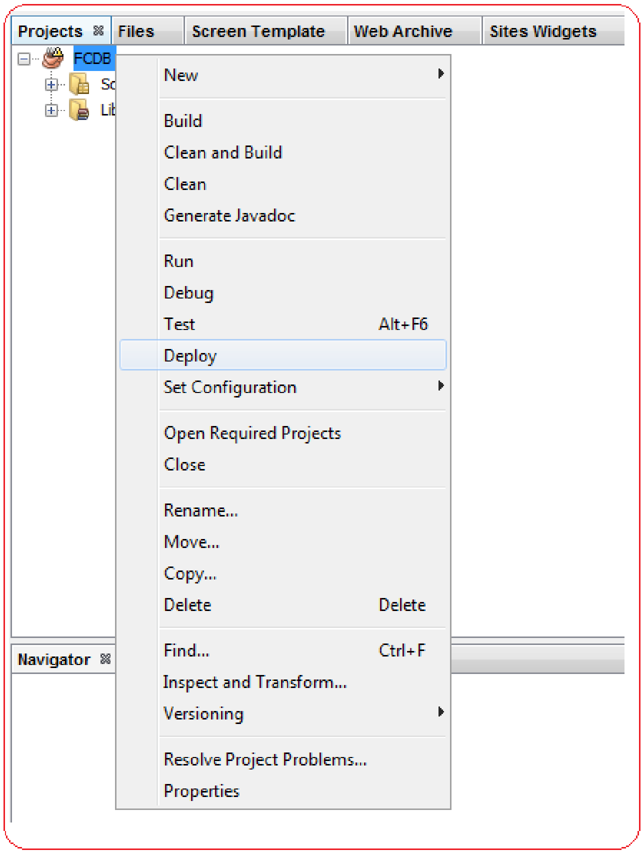


The user can create new Java Class file from FCDB Business Components that contains the existing FCDB classes & interfaces as displayed below:

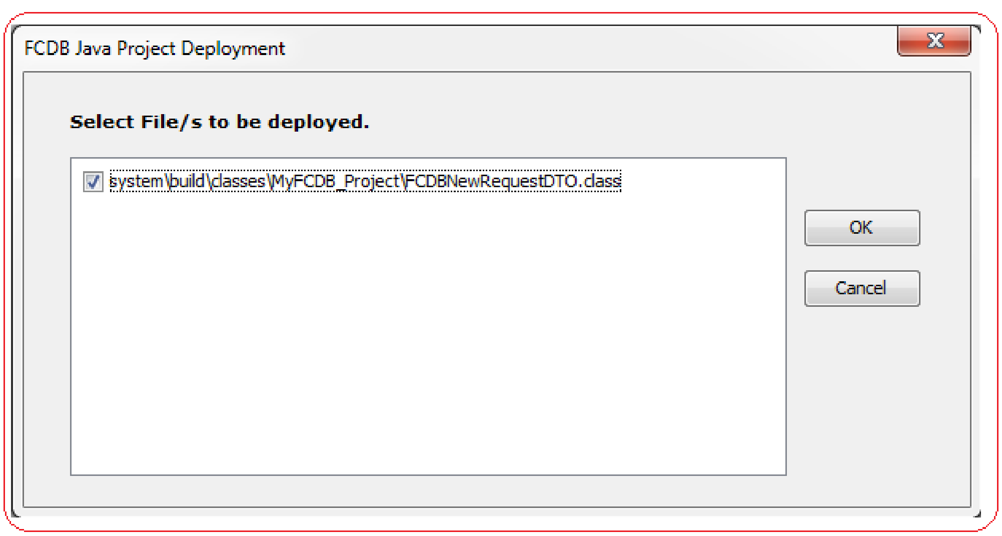


FCDB/Java Project Deployment

The existing Java Class files of FCDB/Java Projects can be deployed on the FCDB server using the tool. To select amongst the list of Java Class files, right click on a project and select Deploy as shown below:

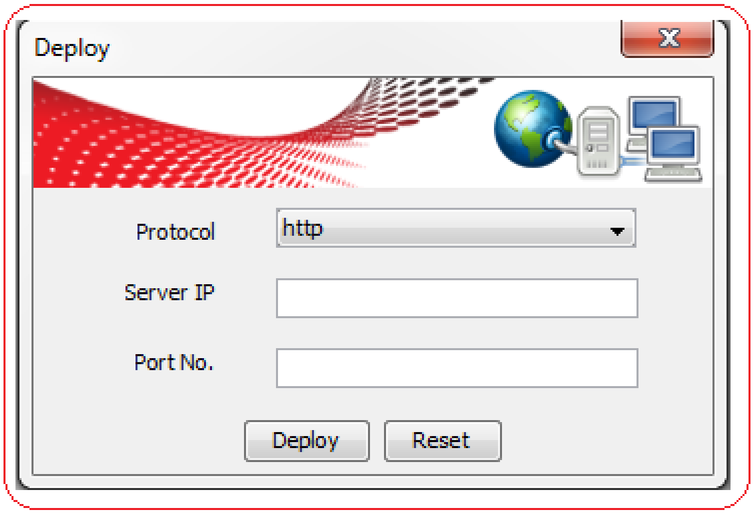


A new window will pop up asking user to select the class file/s to be deployed. Select the files and click on Ok button.



The dialog will show list of all Java Class files in the **system\build\classes** folder of your FCDB setup path. To know more about how to set FCDB properties *click here*.

On clicking OK button, user will be asked for Protocol, Server IP and Port No of FCDB server as shown below:

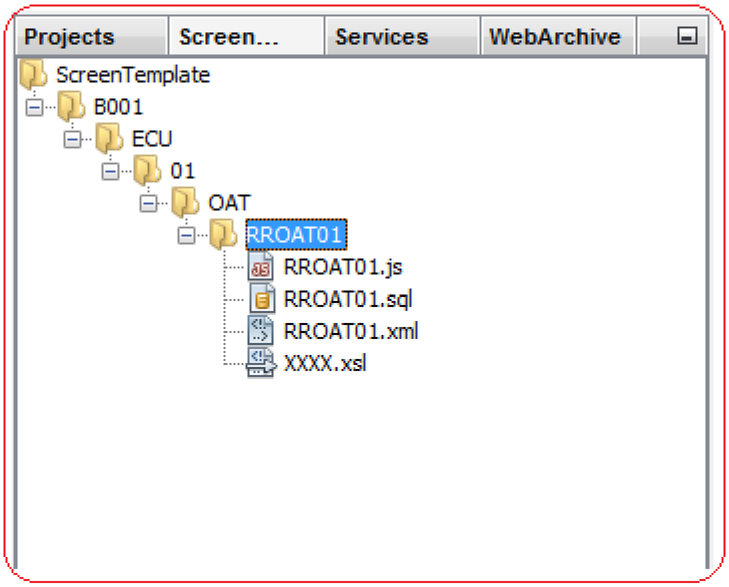


On clicking Deploy button, the selected files will be deployed on the FCDB server.

2. ScreenTemplate Window

a. Screen Design

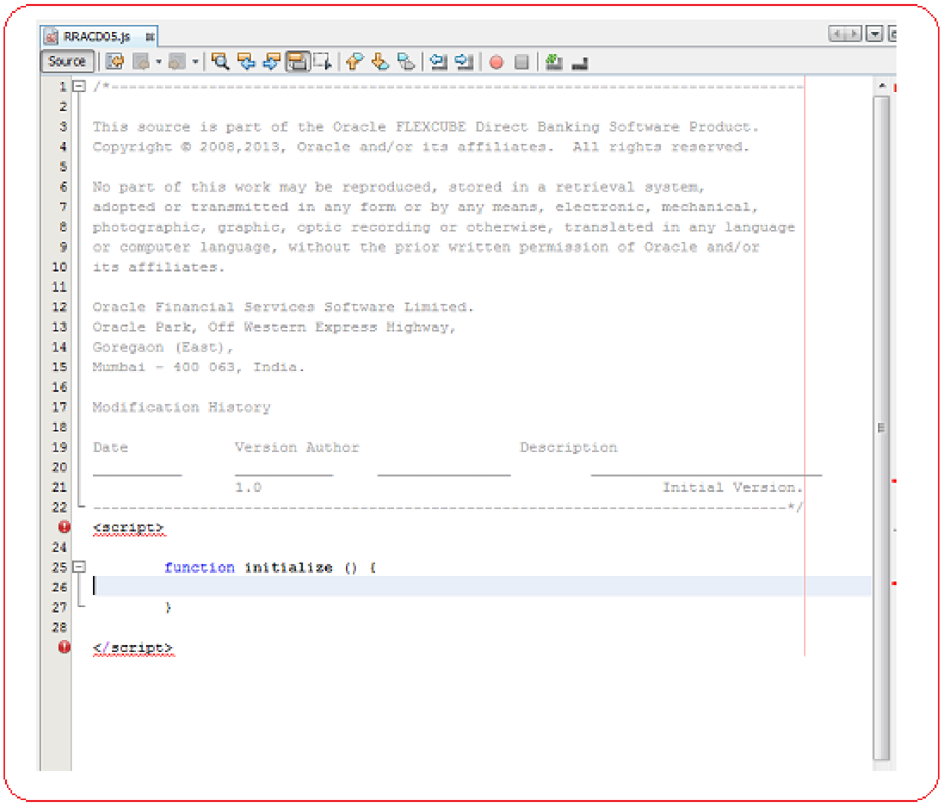
ScreenTemplate window at the topmost left panel of this tool as shown below contains the details of all the ScreenTemplate projects shown in ScreenTemplate Explorer Window which are created in the workspace, in a tree layout format. This Tool creates a folder “ScreenTemplate” in the user’s current directory, this folder is the workspace for this tool. This tree layout represents the actual directory structure created inside the workspace folder.



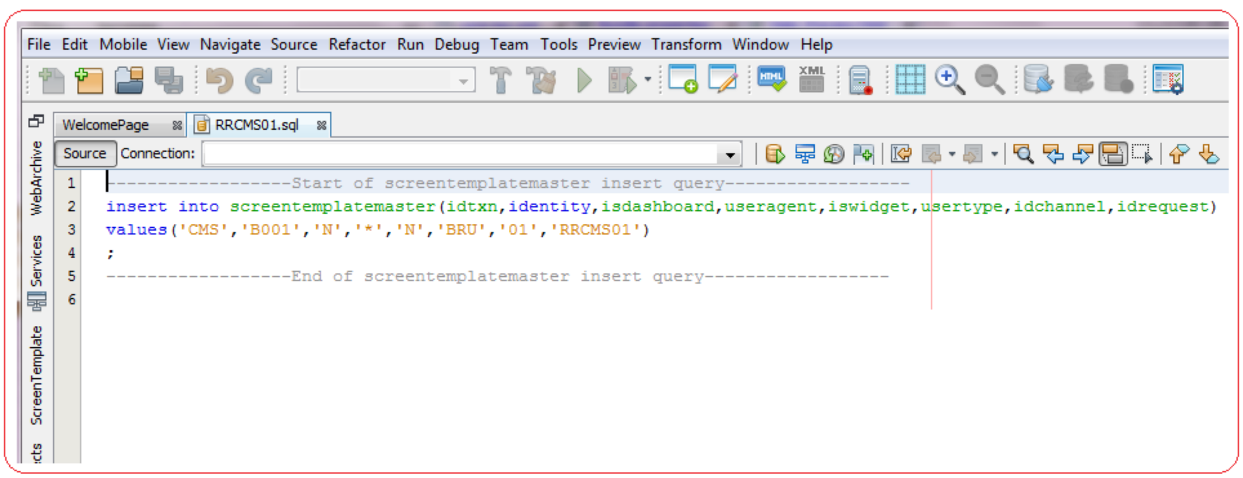
For each ScreenTemplate project there are four files loaded in the workspace.

I. **.js (e.g.RROAT01.js):** Clicking on this tree node will open an editor panel in the center main work area panel as shown below.

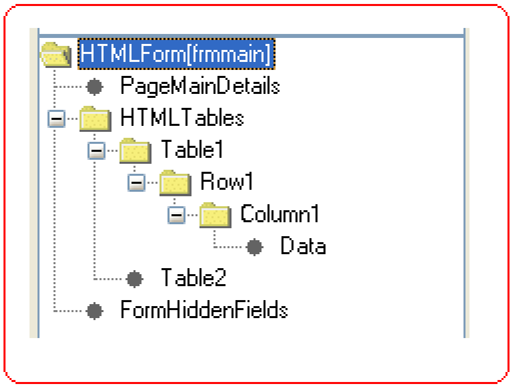
   The content of the “.js” files are loaded in this editor. User can use this editor panel to write/edit the content of the JavaScript file. To save the changes user needs to click the submenu save (Under File Menu) or use “**Ctrl+S”** as keyboard shortcut.



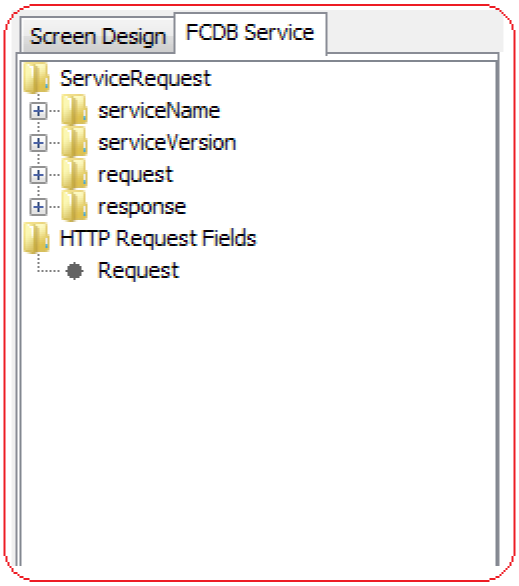
II. **<u>.sql (e.g.RROAT01.sql):</u>** Clicking on this tree node menu opens the script text panel in the center main work area panel as shown below. All the scripts generated for a particular ScreenTemplate project are displayed in this text panel.



III. **.xml (e.g.RROAT01.xml):** Clicking on this tree node loads the HTML Layout Panel, in the bottom left of this tool, with the HTML specific details of the screentemplate as shown below.

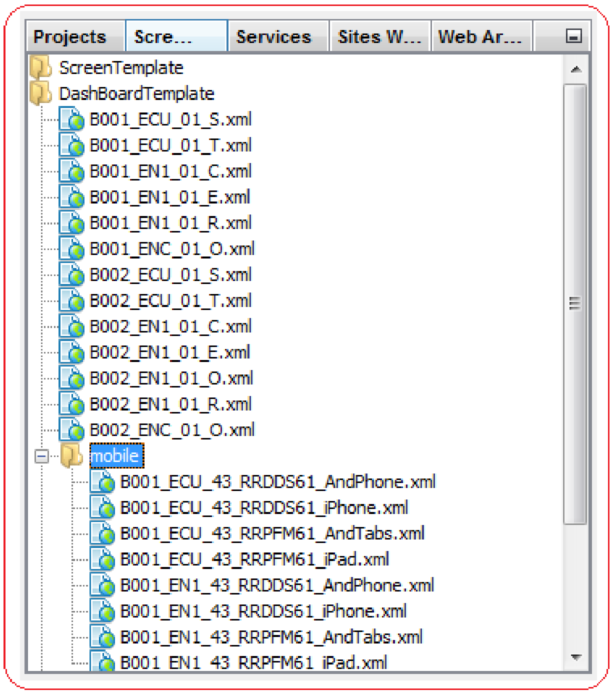


IV. **.xsl (e.g. XXXX.xsl):** Clicking on this tree node loads the FCDB Service Tree Panel, in the bottom left of this tool, with the FCDB Service details of the screen template as shown below.



b. Dashboard Design

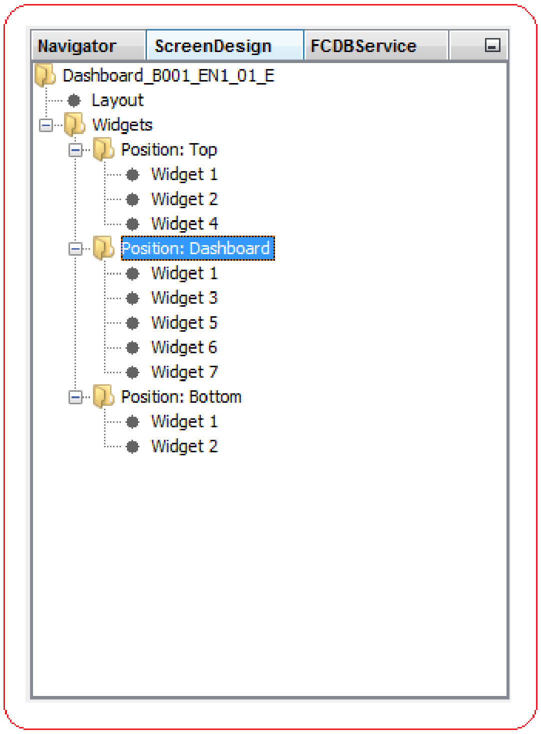
The upper left panel of this tool as shown below contains a tab "ScreenTemplate" to provide a tree based explorer view of all the Screen Design projects (Dashboard and Screen Design) which are created in the workspace.
The IDE creates a folder "DashBoardTemplate" in the directory where the IDE is installed, this folder consists of all the dashboard project xml currently loaded in the IDE.



For each **DashBoard Project (Internet/Mobile)** there is a design xml file loaded in the workspace.

Internet Dashboard Project

Dashboard Design xml:
The name of Internet Dashboard Design xml follows the nomenclature <idEntity>.<Usertype>.<ChannelID>.<DashboardID>.xml. E.g. A Dashboard Project for entity "B001", usertype "EN1" and dashboard ID "E" for Internet Channel has the design xml name as **B001_EN1_01_E.xml**
Double Clicking on this tree node loads the dashboard project into the IDE for the user to design dashboard. A progress bar appears to load the selected project into IDE workspace. On successful loading of project, title of IDE shows the selected project, the Screen Layout Tab in the bottom left of this tool, shows a tree based view of the design xml for the user to edit details of the dashboard as shown below,

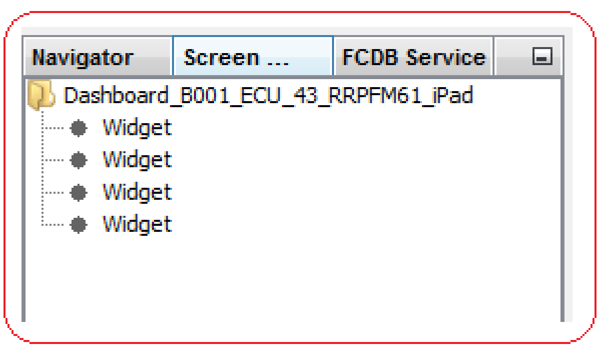


Mobile Dashboard Project

Dashboard Design xml:

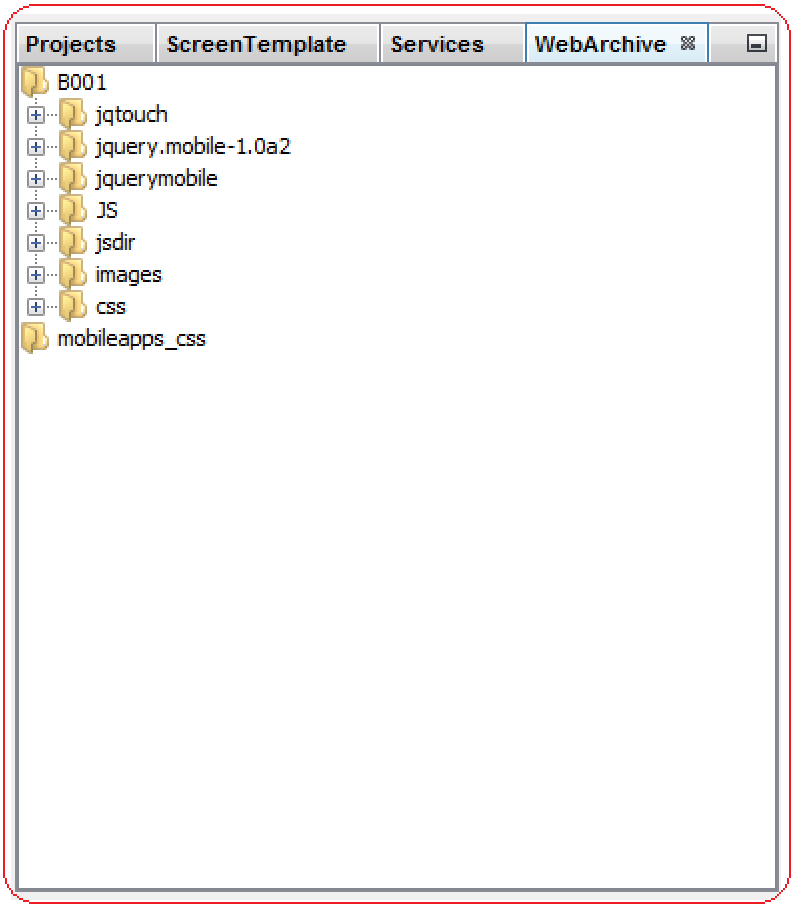
The name of Mobile Dashboard Design xml follows the nomenclature <idEntity>.<Usertype>.<ChannelID>.<DashboardID><.UserAgent>.xml. E.g. A Mobile Dashboard Project for entity "B001", usertype "ECU", dashboard ID "RRPFM61" and user agent "AndTabs" for Mobile Channel has the design xml name as **B001_ECU_43_RRPFM61_AndTabs.xml**

Double Clicking on this tree node loads the mobile dashboard project into the IDE for the user to design dashboard. A progress bar appears to load the selected project into IDE workspace. On successful loading of project, title of IDE shows the selected project, the Screen Layout Tab in the bottom left of this tool shows a tree based view of the design xml for the user to edit details of the dashboard as shown below,



3. WebArchive Window

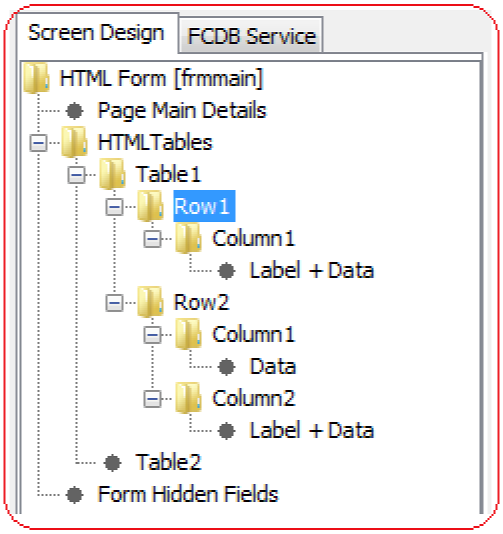
The web archive war (for e.g. B001.war) is loaded from the **preview** folder of the tool into the WebArchive window at the top left corner of the tool as shown below:



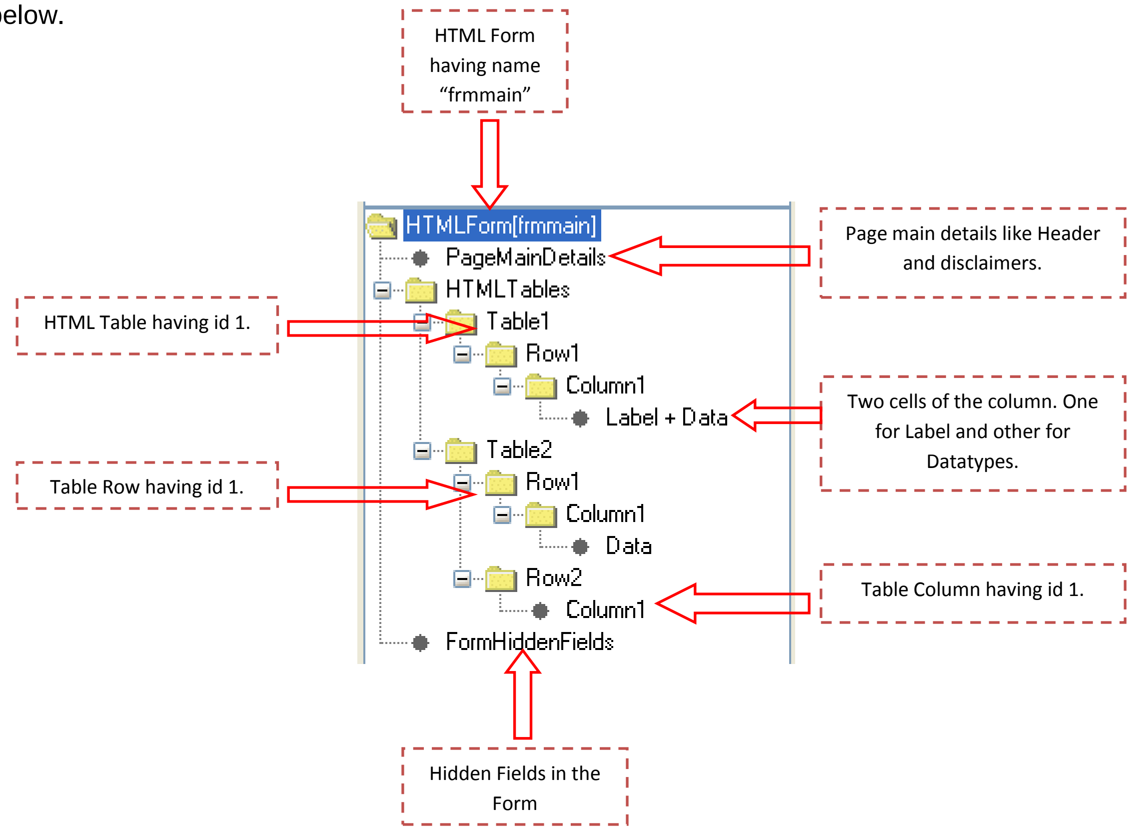
Each time the tool is launched, the preview folder is updated with the latest files extracted from war located in **"deploy/war"** folder of the tool. In order to make changes permanent to the files of web archive, it is necessary to package such files into war (for e.g. B001.war) at location **"deploy/war"** of the tool. If the files are not packaged, the changes will be lost and replaced by old files extracted from war when the tool is launched again. For more details on how to package war, refer to "**Synchronize/Package WAR**" section of the document.

The web archive window lists all the css, js and images files of the war. For more details on how to edit css, js and images files, refer to "**Javascript / CSS / Image Editors**" section of the document.

The HTML Layout Panel opens up on clicking the Screen Design Tab at the bottom-left of this tool as shown below.



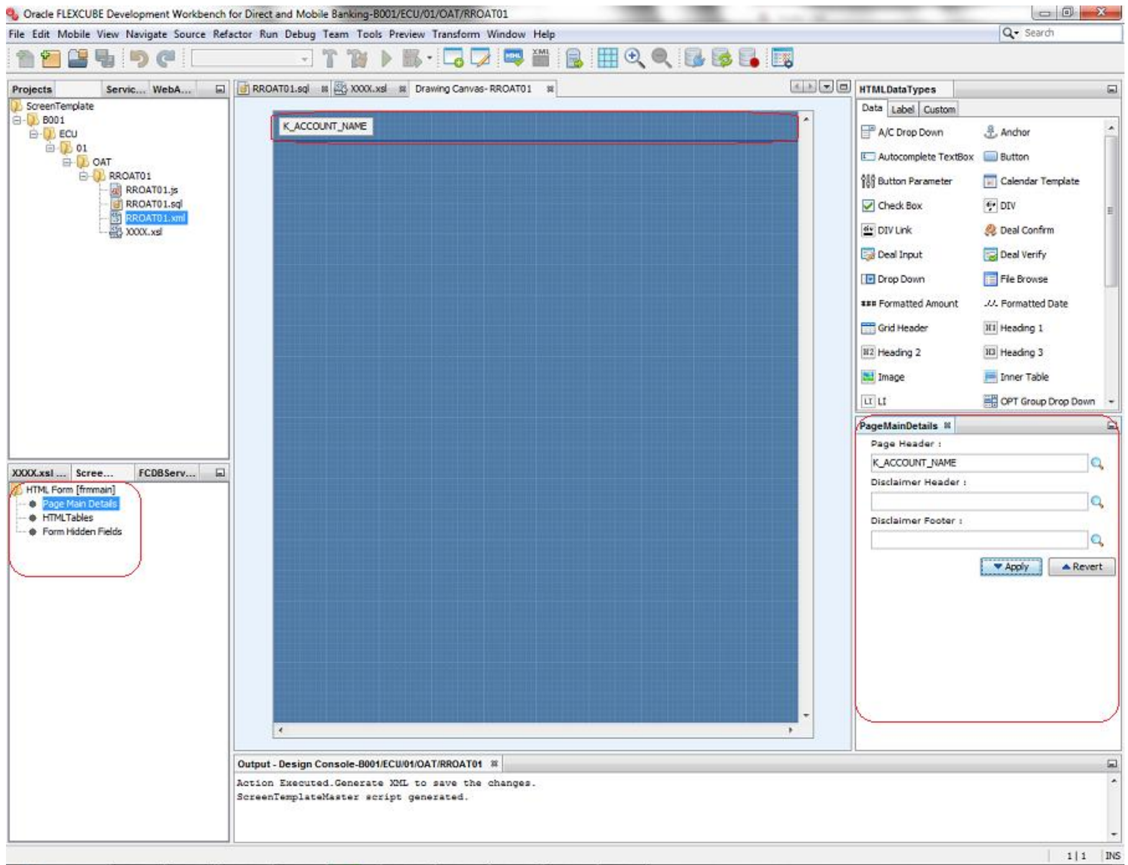
This panel contains the HTML specific details of the screentemplate. This panel loads the screen design in a tree layout format, where each node represents certain HTML component of the screen design as shown below.



Certain nodes shown in the screen shot above have right-click options, user can use these options to add child nodes or delete these nodes. Details of the right-click options available on the nodes are discussed below.

1. Page Main Details

Page Main Details panel appear in the work area, when user clicks on "PageMainDetails" node of HTML Layout panel, as shown below.



Page Main Details helps user to configure following three fields of the page.

I. **Page Header:** User needs to enter XSL keyword (starting with K_) in the text box against this field. The text entered here will appear on the top of the screen design as its page header. A lookup icon is available beside the textbox, clicking this lookup image will open a dialog box where user can search available XSL keywords for a particular text.

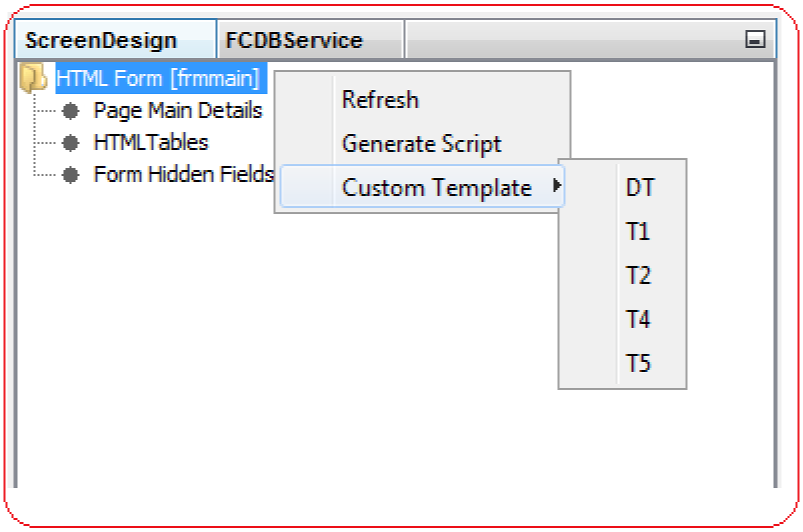
II. **Disclaimer Header:** User needs to enter XSL keyword (starting with K_) in the text box against this field. The text entered here will appear on the top of the screen design as disclaimer header.

III. **Disclaimer Footer:** User needs to enter XSL keyword (starting with K_) in the text box against this field. The text entered here will appear on the bottom of the page as disclaimer footer.

IV. **Total Step Count:** User can set the number of steps that are there in this screen. Current this property is used only for the user agents of Mobile Applications.

2. HTMLForm

User can right-click on this node and will get the following option as shown below.



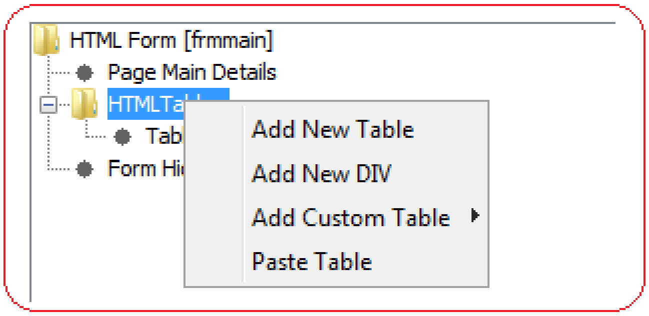
i. **Refresh:** Click on this option it refresh the Layout tree. It means Layout tree is render again and if any table order is changed it reflect after the refresh.

ii. **Generate Script:** Click on this option generate SQL script corresponding to current state of DTO.

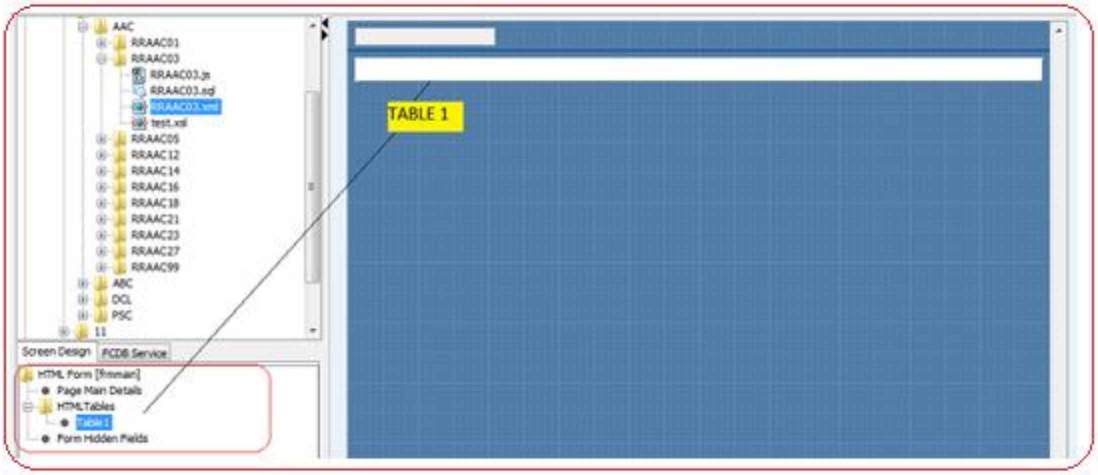
iii. **Custom Template:** Click on these option display custom templates which are available. This feature developed for mobile banking because many of their screens has same layout.

3. HTMLTables

User can right-click on this node and will get the following option as shown below.

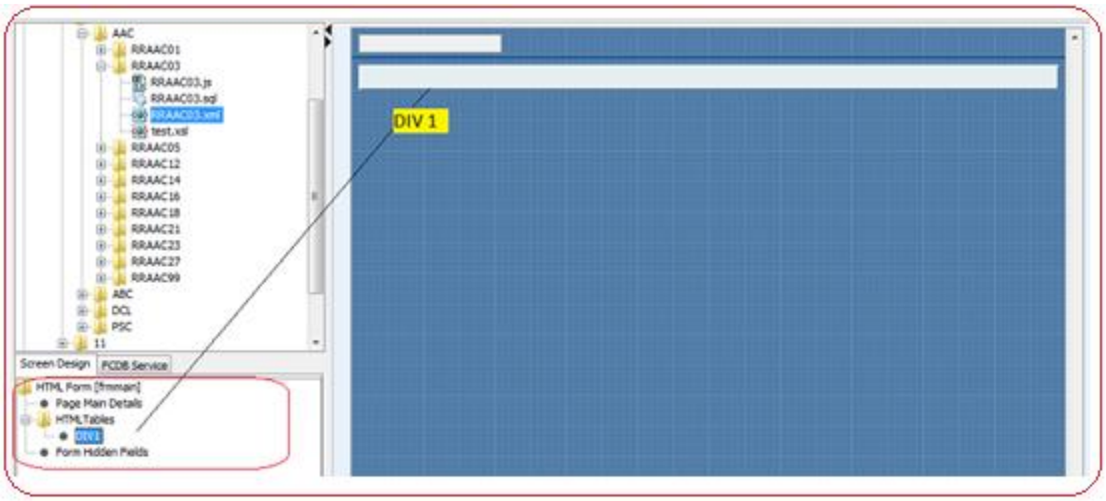


i. **Add New Table:** Click on this option to add a new HTML Table to your screen design. A new Table with a numeric id will automatically get added under the “HTMLTables” node as shown below.

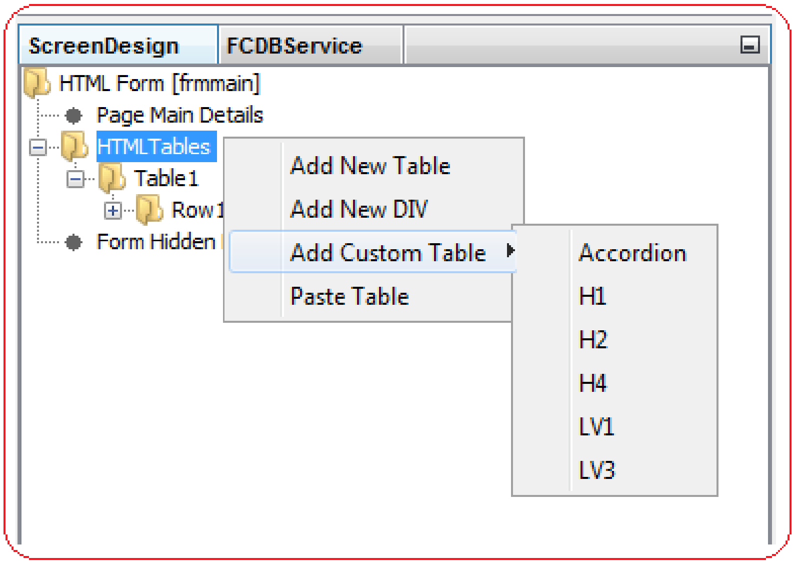


Double-Click this node or Press “F2” to change the numeric id of this newly added table.

ii. **<u>Add New DIV:</u>** Click on this option to add a new HTML DIV to your screen design. A new DIV with a numeric id will automatically get added under the “HTMLTables” node as shown below.



iii. **<u>Add Custom Table:</u>** Click on this option to add a new predefined HTML Table such as Accordion, H1, H2, H4, LV1, and LV3 to your screen design. A predefined table with a numeric id will automatically get added under the “HTMLTables” node as shown below. This feature is available for mobile banking only.

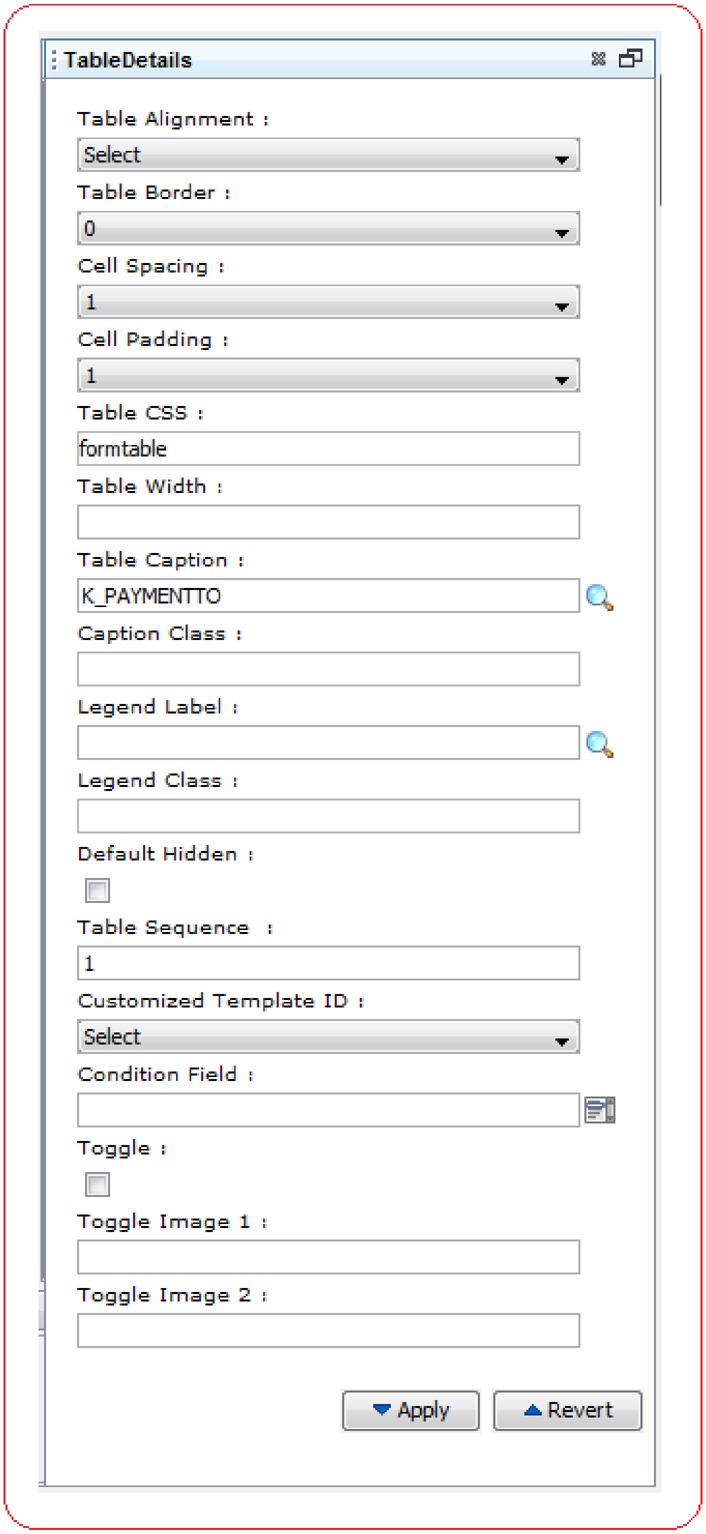


iv. **<u>Paste Table:</u>** Click on this option to add a copied HTML Table to your screen design. This option is available only if user has copied any HTML Table.

v. **<u>Drag and Drop:</u>** Feature to drag and drop a table is also provided in tool. The user can drag and drop a table node from one position to another within HTML Tables to change its Sequence Number.

User may require to configure the properties of the HTML Table like table border, cell spacing, cell padding. To configure these properties user need to click on **"Table/DIV <IdRequest>"** node or double clicking on the Table/DIV painted on canvas.

Clicking this node will display **"Table Details"** window, as shown below. In this panel user will get the fields to configure the HTML properties of the selected HTML table/DIV.


TableDetails
Table Alignment :
Select
Table Border :
0
Cell Spacing :
1
Cell Padding :
1
Table CSS :
formtable
Table Width :
Table Caption :
K_PAYMENTTO
Caption Class :
Legend Label :
Legend Class :
Default Hidden :
Table Sequence :
1
Customized Template ID :
Select
Condition Field :
Toggle :
Toggle Image 1 :
Toggle Image 2 :
Apply
Revert

User can configure following HTML properties of the selected table.

I. **Table Alignment:** User can set the alignment property of the HTML table by entering the text in the text box against this field. Possible values that can be entered are:
   - ✓ left
   - ✓ right
   - ✓ center

II. **Table Border:** User can set the border property of the HTML table by entering the text in the text box against this field. User can enter only numeric values >= 0. Tool will not allow the user to apply any other values. Recommended value for border is between 0 to 5.

III. **Cell Spacing:** User can set the cell spacing property of the HTML table by entering the text in the text box against this field. User can enter only numeric values >= 0. Tool will not allow the user to apply any other values. Recommended value for cell spacing is 1.

IV. **Cell Padding:** User can set the cell padding property of the HTML table by entering the text in the text box against this field. User can enter only numeric values >= 0. Tool will not allow the user to apply any other values. Recommended value for cell padding is 1.

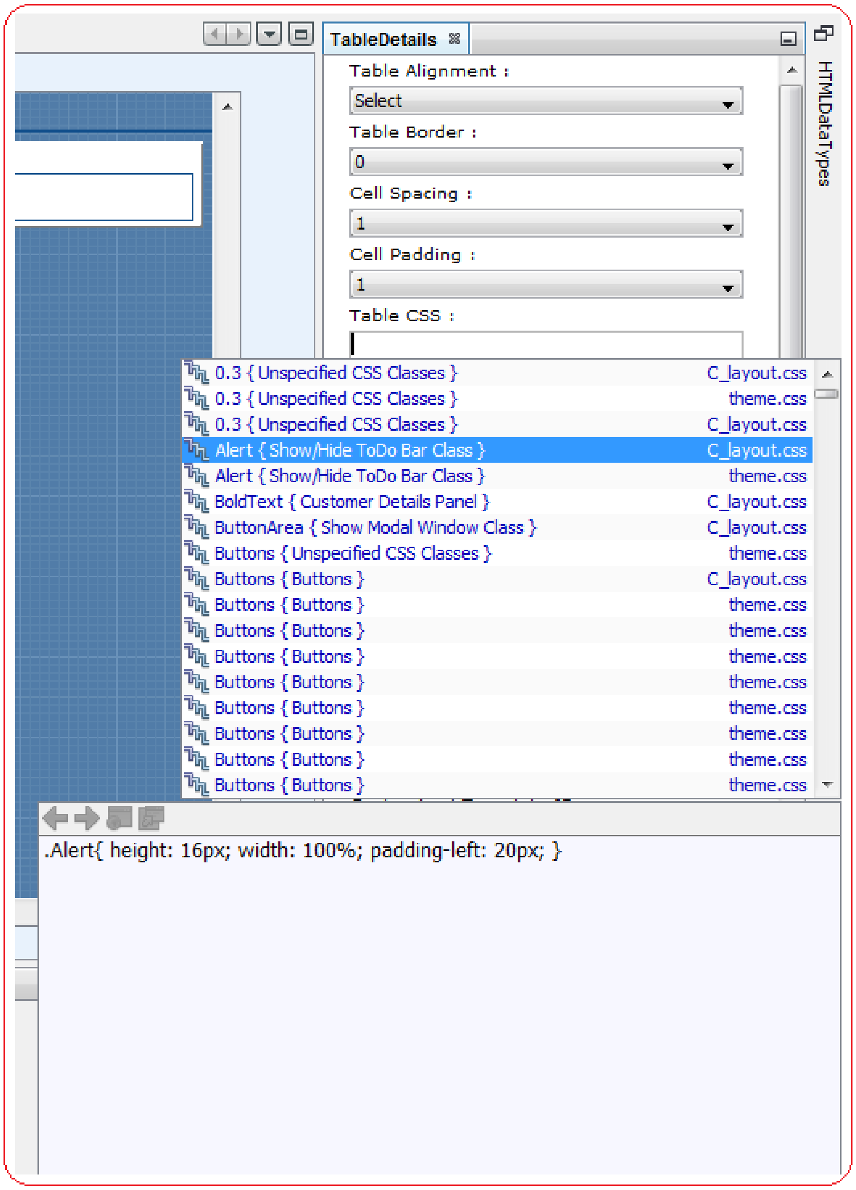
V. **Table CSS:** User can set the class property of the HTML Table by entering the text in the text box against this field. User can only enter the CSS classes which are available in this tool.

   Tool will not allow the user to apply any other CSS class name other than those available in this tool.

   User can use CSS auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available CSS class names as shown below.

   As shown an example of auto-suggest in the below image, the highlighted CSS class displays its class name (E.g. Alert) and group name (E.g. Show/Hide ToDo Bar Class) to the left and its corresponding css file name (E.g. L_layout.css) to the right. The documentation window below displays its respective CSS code.

   User can also press **Ctrl+P** to check its CSS code as Tool Tip.



Recommended CSS classes for HTML tables are:

1. **formtable:** Use this class when the table contains certain input fields.
2. **formtablereadonly:** Use this class to display the table on verification screen where no input fields are there.
3. **infotable:** Use this class to display certain info inside a table.
4. **gridtable:** Use this class for table in grid format.
5. **uidownload:** Use this class for calling UI download framework of FCDB.
6. **buttonarea:** Use this class for making form submission buttons.
7. **tabpanel:** Use this class for making set of tabs inside a table.

VI. **Table Width:** User can set the width property of the HTML table by entering the text in the text box against this field. User can enter numeric value with "px" or "%" as postfix. Tool will not allow the user to apply any other values. Recommended value for width is, it can be left blank or 100%.

VII. **Table Relative Width (For User Agents of Mobile Applications):** User can set the width of the table relative to the width of entire screen i.e. Width of the Table will be calculated as % of the total width of the screen. User can enter only decimal values and value 1.0 represents the entire screen width.

VIII. **Table Relative Height (For User Agents of Mobile Applications):** User can set the height of the table relative to the height entire screen i.e. Height of the Table will be calculated as % of the total height of the screen. User can enter only decimal values and value 1.0 represents the entire screen height.

IX. **Table Relative Position X (For User Agents of Mobile Applications):** User can set the X-Coordinate of the table relative to the entire screen i.e. X-Coordinate of the Table will be calculated as % of the total width of the screen. User can enter only decimal values.

X. **Table Relative Position Y (For User Agents of Mobile Applications):** User can set the Y-Coordinate of the table relative to the entire screen i.e. Y-Coordinate of the Table will be calculated as % of the total height of the screen. User can enter only decimal values.

XI. **Table Caption:** User can set the caption of the HTML table by entering the text in the text box against this field. User can enter only XSL keyword (starting with K_) in this text box. Tool will not allow the user to apply values not in the format of XSL keyword.

XII. **Caption Class:** User can set the class property of the Table Caption. User can only enter the CSS classes which are available in this tool. User can use CSS auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available CSS class names as mentioned above. There is no recommended value for this field, it can be left blank.

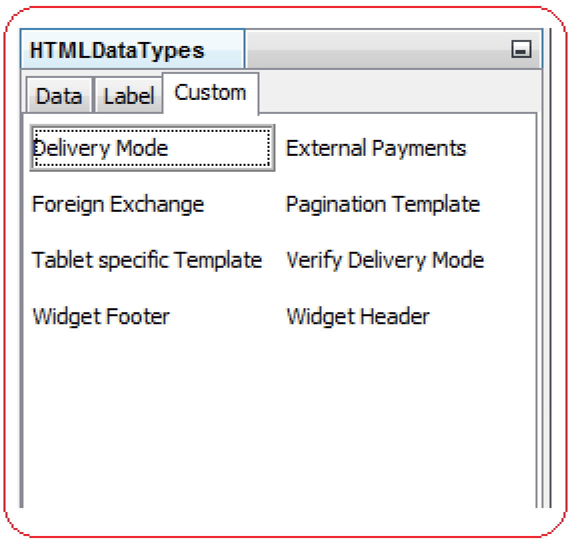
XIII. **Legend Label:** User can create a legend for the HTML Table by entering the text in the text box against this field. User can enter only XSL keyword (starting with K_) in this text box. Tool will not allow the user to apply values not in the format of XSL keyword. Legends are recommended only for inner HTML tables.

XIV. **Legend Class:** User can set the class property of the table legend. User can only enter the CSS classes which are available in this tool. User can use CSS auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available CSS class names as mentioned above. Recommended value for this field is "labeltext" or it can be left blank.

XV. **Default Hidden:** This check box can be checked if the user wants to make this HTML table default hidden when the screen output html page loads for the first time. This feature is recommended where certain HTML table in the form needs to be kept hidden and displayed only on some JavaScript event.

XVI. **Table Sequence:** User can set the sequence number of the HTML table by entering a numeric value in the text box against this field. User can enter only numeric value > 0. This property helps the user to rearrange the order in which the HTML Tables will appear in the form. For e.g., if the current sequence number of the table is 3, it will appear as third HTML table in the form.

XVII. **Customized Template ID:** User can use already existing custom templates available as custom datatypes in this panel. User can select any custom template available in the dropdown list. Use this feature only when you want to use an existing HTML Table layout available, else leave this field blank. User can refer to the data types available in the HTMLDatatypes panel grouped under the "Custom" tab. User can click on the data type button and value will get selected in the drop down against this field.



XVIII. **Condition:** This property contains the display condition of a table. User can set the conditions basis on which the table will get displayed or will not get displayed. This property is only valid for mobile banking. An enlarge icon is available beside the textbox, clicking this enlarge image will open a dialog box where user can enter the text in enlarged text editor.

XIX. **Display Type:** This property contains the default display type of a table. This property is available only for the screen design of user agents (IPhone, IPad, Android Tab, Android Phone) of Mobile Applications. User can select any Value Normal, Hidden, Modal, Popup or Wrapper from the list.

XX. **Is Scrollable:** This property indicates whether the table is scrollable or not.

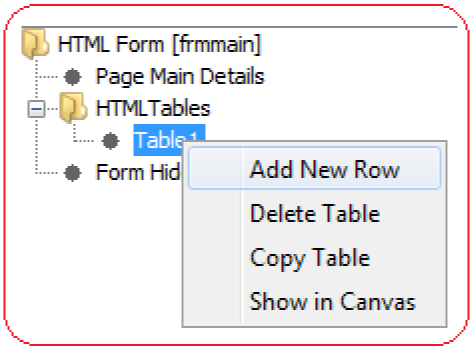
XXI. **Toggle:** This property indicates whether toggle functionality is available on table or not.

XXII. **Toggle Image 1:** This field indicates relative path for toggle image 1.

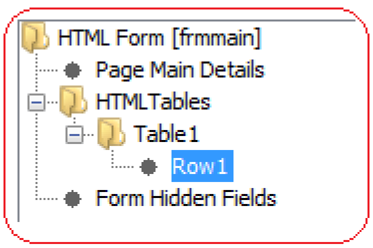
XXIII. **Toggle Image 2:** This field indicates relative path for toggle image 2.

4. HTML Table Rows

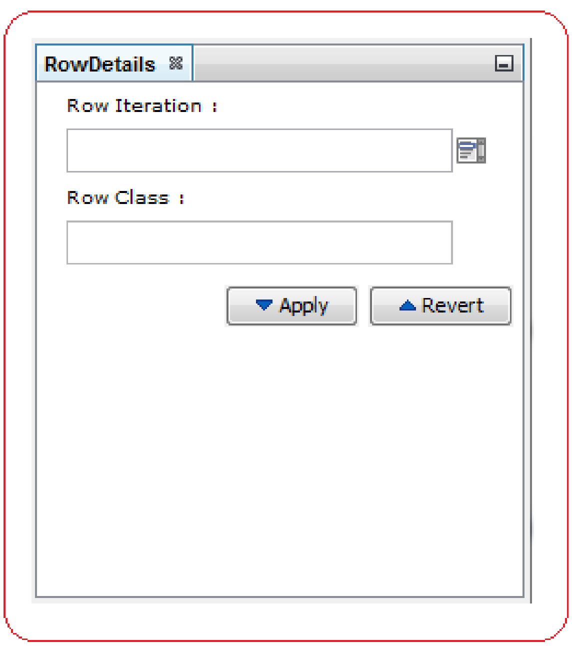
For adding rows to the table, right click on Tables tree node. A menu appears with following option.



i. **Add New Rows:** Click on this option to add a new row to your HTML Table. A new row with numeric id will be automatically added under the current HTML Table as shown below,



Double-Click this node or Press "F2" to change the numeric id of this newly added table row. Clicking on the row node added shall open a property palette at the right hand side of tool to allow user to edit the row level properties as shown below,



1. **Row Iteration:** User can enter the XPATH on which they can iterate to create multiple rows dynamically or to create an array of Hidden fields.

2. **Row Class:** User can enter the css class to be applied at row level. User can only enter the CSS classes which are available in this tool. User can use CSS auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available CSS class names.

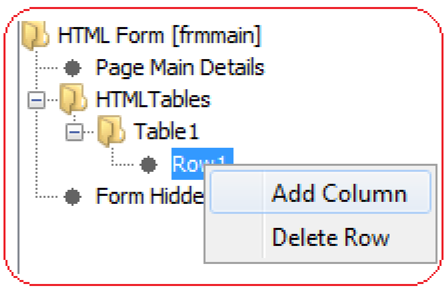
ii. **Delete Table:** Click on this option to delete the selected HTML table. The current HTML Table will get deleted along with its child nodes, if any.

iii. **Copy Table:** Click on this option to copy this HTML Table. Once copied user will get the right click option to paste the copied table in **"HTMLTables"** node.

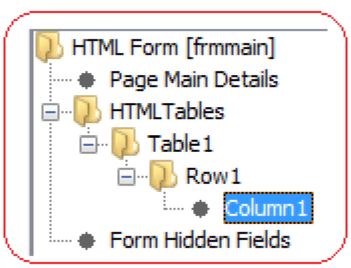
iv. **Show in Canvas:** Click on this option to show this HTML Table on canvas.

5. HTML Table Columns

For adding Column to the table right click on Rows tree node. A menu appears with following option:



i. **Add Column:** Click on this option to add a new column to your HTML Table Row. A new column with numeric id will automatically get added under the current HTML Table Row as shown below.



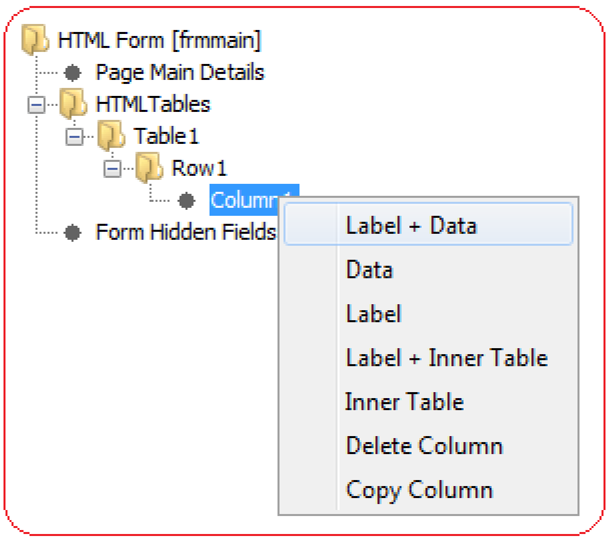
Double-Click this node or Press "F2" to change the numeric id of this newly added column.

ii. **Delete Row:** Click on this option to delete the selected row. The current row will get deleted along with its child nodes, if any.

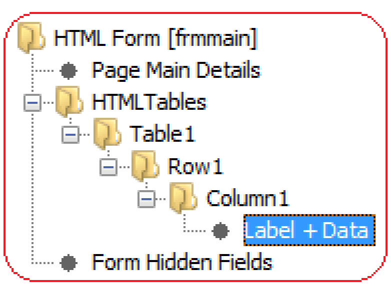
6. HTML Table Column Components

For adding columns to the table row, right click on column tree node. A menu list appears with following 7 options:

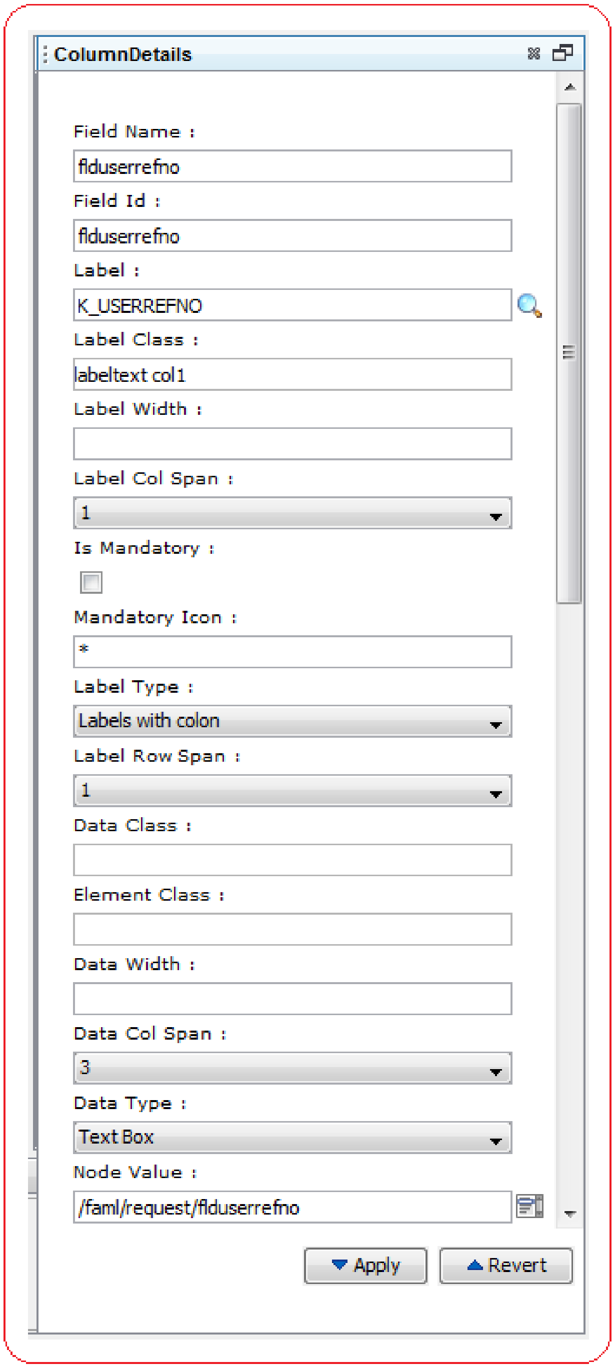
- ✓ Label+Data
- ✓ Data
- ✓ Label
- ✓ Label+Inner Table
- ✓ Inner Table
- ✓ Delete Column
- ✓ Copy Column



The selected menu item gets added under Column Node. Selecting Delete Column will delete the Column Node. Selecting Copy Column will copy the current Column.

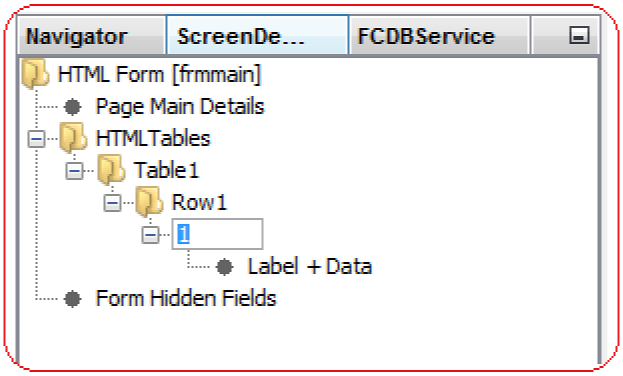


Clicking on the node added shall open a palette at the right hand side of tool to allow user to edit the properties HTML Table Data as shown below,



On the basis of type of node added, a combination of input fields discussed below will appear in the panel. Description of fields is given below:

i. **Column ID:** Each Component added to a table in Screen Design is given a unique ID. This ID is automatically generated by the tool and is appended in the name of that component node in Screen Layout Tree. To Change the Column ID of a component, double click on its node in Screen Layout Navigator Tree in lower left side of the tool. The Node shall become editable as shown below,



ii. **<u>Drag and Drop:</u>** Feature to drag and drop a column node in also provided in the tool. The user can drag and drop a column node from one position to another within different rows to change its column ID.

iii. **<u>Field Name:</u>** User has to enter a field name. It is a mandatory field and cannot be left blank. It is always recommended to use "fld" as prefix with the field name, for e.g. fldbeneficiary. User can enter an alphanumeric value for Name.

iv. **<u>Field ID:</u>** User has to enter a field id. It is a mandatory field and cannot be left blank. This field id has to be unique; user cannot use any field id which has been used while defining the screen. It is always recommended to use "fld" as prefix with the field id, for e.g. fldbeneficiaryid. User can enter an alphanumeric value for Id.

v. **<u>Component Relative Width (For User Agents of Mobile Applications):</u>** User can set the width of the component's relative to the width of entire screen i.e. Width of the component will be calculated as % of the total width of the screen. User can enter only decimal values and value 1.0 represents the entire screen width.

vi. **<u>Component Relative Height (For User Agents of Mobile Applications):</u>** User can set the height of the Component relative to the height entire screen i.e. Height of the Component will be calculated as % of the total height of the screen. User can enter only decimal values and value 1.0 represents the entire screen height.

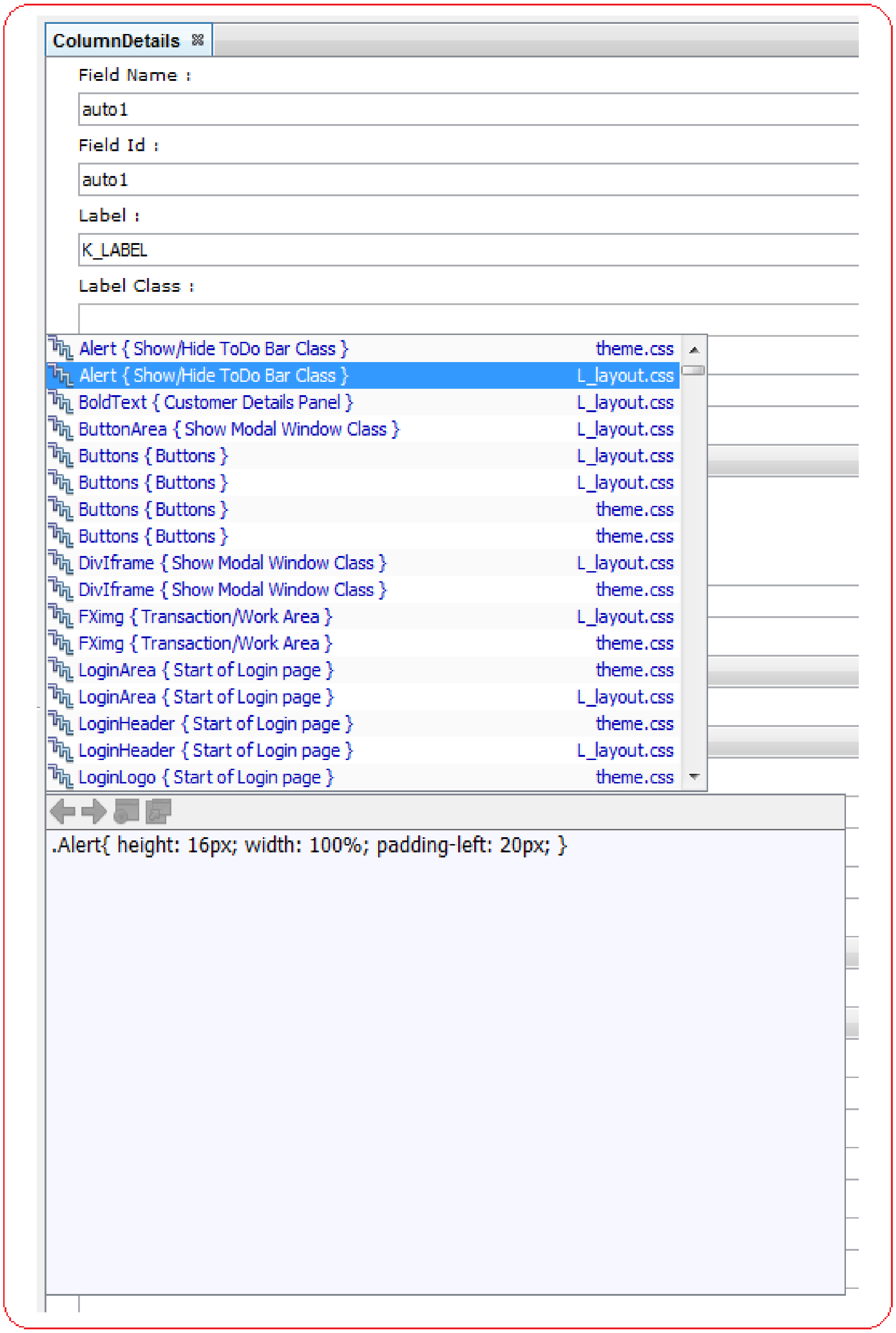
vii. **<u>Component Relative Position X (For User Agents of Mobile Applications):</u>** User can set the X-Coordinate of the Component relative to the table in which it is enclosed i.e. X-Coordinate of the Component will be calculated as % of the total width of the table. User can enter only decimal values.

viii. **<u>Component Relative Position Y (For User Agents of Mobile Applications):</u>** User can set the Y-Coordinate of the Component relative to the table in which it is enclosed i.e. Y-Coordinate of the Component will be calculated as % of the total height of the table. User can enter only decimal values.

ix. **<u>Label:</u>** User can enter XSL keyword (starting with K_) in the text box against this field. This text will appear as a label in the first cell of the column. Tool will not allow the user to apply values not in the format of XSL keyword.

x. **Label Class:** User can enter the css class to be applied on the text entered against label. User can only enter the CSS classes which are available in this tool.

User can use CSS auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available CSS class names as shown below.



As shown an example of auto-suggest in the above image, the highlighted CSS class displays its class name (E.g. Alert) and group name (E.g. Show/Hide ToDo Bar Class) to the left and its corresponding css file name (E.g. L_layout.css) to the right. The documentation window below displays its respective CSS code. User can also press **Ctrl+P** to check its CSS code as Tool Tip. Tool will not allow the user to apply any other CSS class name other than those available in this tool.

Recommended CSS classes for labels are:

- ✓ **labeltext:** Use this class to keep the label text right aligned.
- ✓ **col1:** Use this class for giving a width of 30% to the cell containing the label.
- ✓ **col3**: Use this class for giving a width of 20% to the cell containing the label.
- ✓ **col4:** Use this class for giving a width of 30% to the cell containing the label.
- ✓ **col80:** Use this class for giving a width of 80% to the cell containing the label.

  One can use a combination of these classes by giving a space between the classes, for e.g. "labeltext col1" will make the label right aligned with a width of 20%.

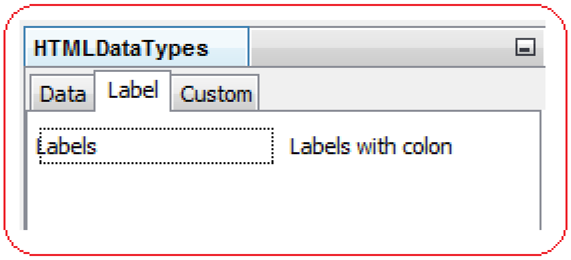
xi. **Label Width:** User can set the width property of the cell containing the label text by entering the text in the text box against this field. User can enter numeric value with "px" or "%" as postfix. Tool will not allow the user to apply any other values. It is recommended to use the CSS classes, as discussed above, to handle the width and leave this field blank. If the above recommended classes do not match your width requirement, then use this field to set the width.

xii. **Label Col Span:** User can enter a numeric value (>0) in the text box against this field. This value will determine the number of columns this cell, containing the label, will span.

xiii. **Is Mandatory:** User has to check this check-box, if they want to append the mandatory icon, discussed below, to the label of this field.

xiv. **Mandatory Icon:** User can enter the key-board special characters like "**" to mark this field as mandatory/or can append this icon to the label for any disclaimer note.

xv. **Label Type:** User can select the label types available in this drop down list. User can refer to the label types available in the HTMLDatatypes panel grouped under the "Label" tab. User can click on the data type button and value will get selected in the drop down against this field, as shown below:



xvi. **Label Row Span:** User can enter a numeric value (>0) in the text box against this field. This value will determine the number of rows this cell, containing the label, will span.

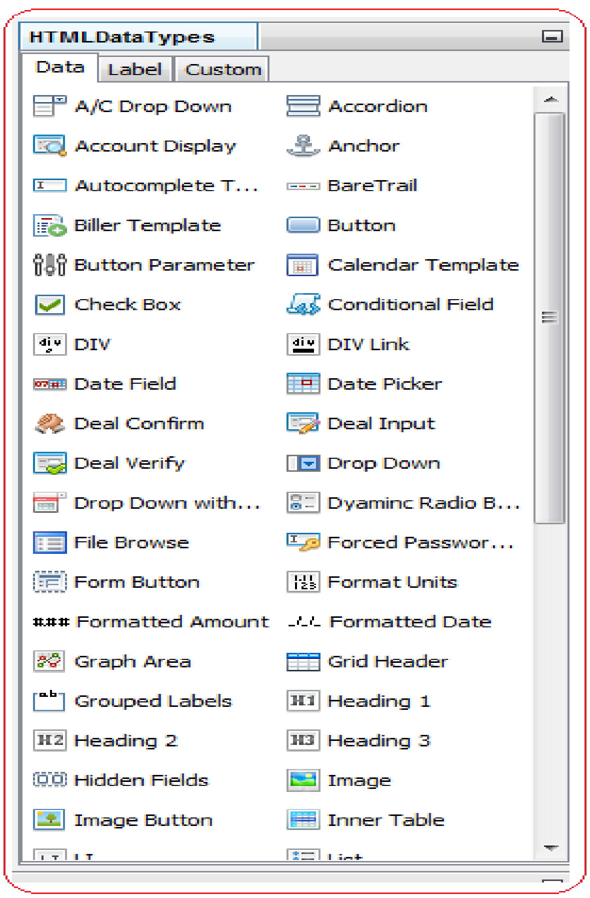
xvii. **Data class:** This CSS class will be applied on the column cell containing the HTMLDatatypes like textbox. User can use CSS auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available CSS class names as mentioned above.

xviii. **Element class:** This CSS class will be applied on the Datatype elements like Textbox, Checkbox, etc. If this field is empty default classes shall be applied on elements. User can use CSS auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available CSS class names.

xix. **Data Width:** This width will be applied on the column cell containing the HTMLDatatypes like textbox. For details refer to Label Width.

xx. **Data Col Span:** User can enter a numeric value (>0) in the text box against this field. This value will determine the number of columns this cell, containing the HTMLDatatypes, will span.

xxi. **Data Type:** User can select the data type available in this drop down list. User can select the HTMLDatatypes to be displayed in the cell of the column. User can refer to the data types available in the HTMLDatatypes palette grouped under the "Data" tab. User can click on the data type button and value will get selected in the Datatype drop down.
User can also drag and drop the Datatype on the corresponding cell in the drawing canvas.



xxii. **Node Value:** User can use text box against this field to populate data in different HTMLDatatypes. For e.g. user can enter a XPATH or a constant value (string'<value>') in textbox HTMLDatatypes to populate the textbox with a default value. Similarly user can use this field to populate list of values in drop down.

xxiii. **Function Name:** User can give the JavaScript function name which they want to attach to the main event of the HTMLComponent, for e.g. giving a function name for dropdown will assign this function to onchange event of the dropdown. The logic of the function has to be written in the corresponding idrequest .js file.

xxiv. **Function Argument:** User can give comma separated arguments to the JavaScript function, discussed above. User can pass only constant values using this field.

xxv. **Input Size:** This field is used for different datatypes with different purposes.

- ✓ **Textbox**: Size of the textbox.
- ✓ **Textarea:** Rows value of the textarea.
- ✓ **Dropdowns:** Making a dropdown into a list.
- ✓ **Radio Button:** Number of radio buttons.
- ✓ **Tabs:** Number of tabs.
- ✓ **Image:** Height of the image.
- ✓ **Buttons:** Size of the button.

xxvi. **Input Maximum Length:** This field is used for different datatypes with different purposes.

- ✓ **Textbox:** Maximum Length of the Textbox.
- ✓ **Textarea:** Columns value of textarea.
- ✓ **Image:** Width of the image.
- ✓ **Buttons:** Maximum Length of the buttons.

xxvii. **UDF Field:** "isudf" allows a user to pass the field value in UDFDTO array object to the service, small code snippet in the corresponding service xsl

```
<xsl:if test="/faml/request/fldudf">
<udfFields>
<xsl:for-each select="/faml/request/fldudf">
<UDFDTO>
<xsl:variable name="udfname"><xsl:value-of select="concat ('/faml/request/',
.)"/></xsl:variable>
<udfName><xsl:value-of select="."/></udfName>
<udfValue><xsl:value-of select="dyn:evaluate($udfname)"/></udfValue>
</UDFDTO>
</xsl:for-each>
</udfFields>
</xsl:if>
```

xxviii. **Alternate Text:** User can enter XSL keyword (starting with K_) in the text box against this field. This text will appear as a tooltip for the HTMLDatatypes. Tool will not allow the user to apply values not in the format of XSL keyword.

For dynamic tooltip user can enter XPATH in this field. For combination of static and dynamic tooltip one should use **"^"** as the delimiter, so for an input like *"K_ACCOUNT^K_BLANK^K_IS^K_BLANK^/faml/response/preparerepaymentiqresponsedto/custloanaccounts/customeraccountdto/accounts/accountnodto [position()=$rowiteration]/nbraccount",*
tooltip will appear as "Account is (evaluated account no)".

xxix. **Image URL:** User can enter relative path of the image file to be displayed. Recommended to be used when one is using "Image" or 'Text Box with Lookup" as HTMLDatatypes. Relative path of commonly used images are:

- ✓ **images/eng/calander.gif:** Use this path for a calendar image.
- ✓ **images/lookup.gif:** Use this path for lookup image.

xxx. **Default Static Label:** User can enter XSL keyword (starting with K_) in the text box against this field. This field is used for different datatypes with different purposes.

- ✓ **Drop Downs:** First static option K_SELECT in the dropdown list, if required.
- ✓ **Static drop downs:** "~" separated static options to be displayed in the dropdown list.
- ✓ **Radio buttons:** "~" separated values to be assigned as label to each radio button.
- ✓ **Image:** Defines the value of alt attribute of image.
- ✓ **Buttons:** Label to be rendered on button.
- ✓ **Anchors:** Text to be rendered as link.

xxxi. **Is Readonly:** User can check this check-box to make the field read-only. For e.g. a read-only text box.

xxxii. **Tab Index:** User can enter a numeric value (>0) as tab index for this field.

xxxiii. **Default Value:** This field is used for different datatypes with different purposes.

- ✓ **Textarea:** Textarea wrap attribute value.
- ✓ **Dropdowns:** Default selected value of the dropdown. One can give XPATH or a constant value like string ('<value>').
- ✓ **Radio Buttons:** This value will determine which radio button will be default selected.
- ✓ **Check Box:** Check box "checked" attribute value.
- ✓ **Image:** Image align attribute value.
- ✓ **Anchor:** Anchor target attribute value.

xxxiv. **Data Row Span:** User can enter a numeric value (>0) in the text box against this field. This value will determine the number of rows this cell, containing the HTMLDatatypes, will span.

xxxv. **Dynamic Function Arguments:** User can give comma separated XPATH's as arguments to the JavaScript function. These XPATH will be evaluated by the framework and their value will always come after the constant function arguments, discussed above in "Function Arguments". For e.g. if function name is "fnViewLoanDetails", function arguments are "'LDT', '01'" and dynamic function arguments are *"//faml/response/prepareloandetailsresponsedto/custaccounts/customeraccountdto[1]/accounts/accountnodto[position()=$rowiteration]/nbraccount, //faml/response/prepareloandetailsresponsedto/custaccounts/customeraccountdto[1]/account*

s/accountnodto[position()=$rowiteration]/nbrbranch", then function fnViewLoanDetails will have arguments as ('LDT', '01', 'evaluated account no', 'evaluated branch').

xxxvi. **Condition:** This property contains the display condition of a field. User can set the conditions basis on which the field will get displayed or will not get displayed. For e.g., *str:split(//faml/request/cmbpaytype,"~")[1]!=2*. This property is only valid for mobile banking.

xxxvii. **Step Number:** User can set the step number of the screen on which this component has to be shown. Currently this field is used for user agents of mobile applications.

xxxviii. **Token 1:** This is a configuration field and can be used for different datatypes for different purposes.

xxxix. **Token 2:** This is a configuration field and can be used for different datatypes for different purposes.

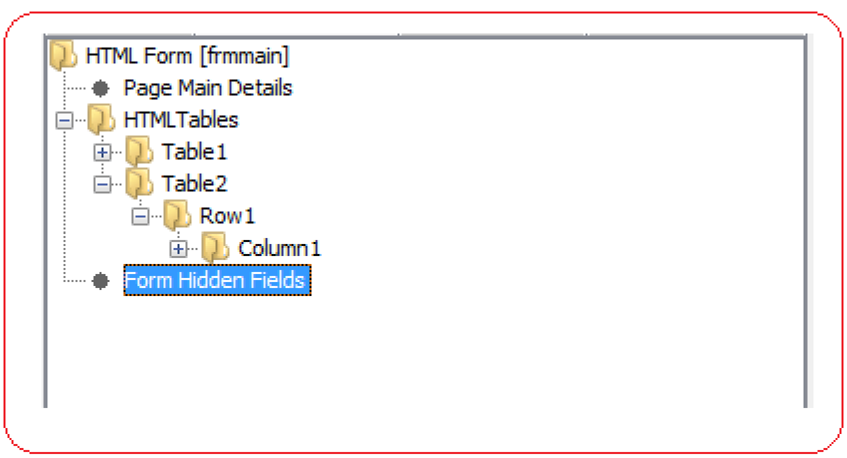
xl. **Token 3:** This is a configuration field and can be used for different datatypes for different purposes.

xli. **Token 4:** This is a configuration field and can be used for different datatypes for different purposes.

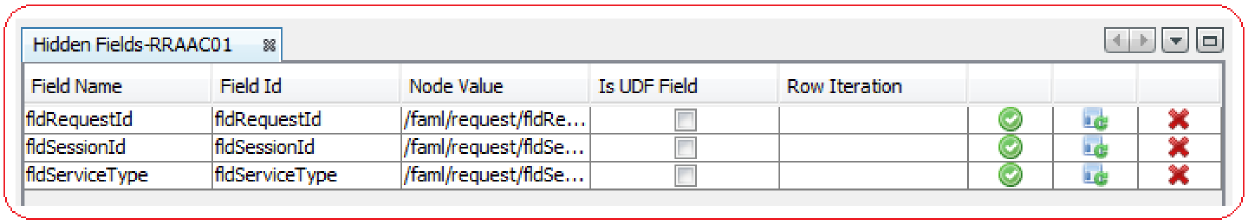
xlii. **Token 5:** This is a configuration field and can be used for different datatypes for different purposes.

7. Form Hidden Fields

In a HTML Form, user may require to pass some hidden fields. To add hidden fields user need to click on the "FormHiddenFields" node as shown below.



Clicking this node will display "Form Hidden Field" panel in the Main Work Area, as shown below. In this panel user will be able to configure the HTML Form Hidden Fields. User can add, delete and modify hidden fields.



As shown in the screen shot above the hidden fields are displayed in a table grid format with following columns:

i. **Field Name:** User has to enter a field name. It is a mandatory field and cannot be left blank. It is always recommended to use "fld" as prefix with the field name, for e.g. fldbeneficiary.

ii. **Field Id:** User has to enter a field id. It is a mandatory field and cannot be left blank. This field id has to be unique; user cannot use any field id which has been used while defining the screen. It is always recommended to use "fld" as prefix with the field id, for e.g. fldbeneficiaryid.

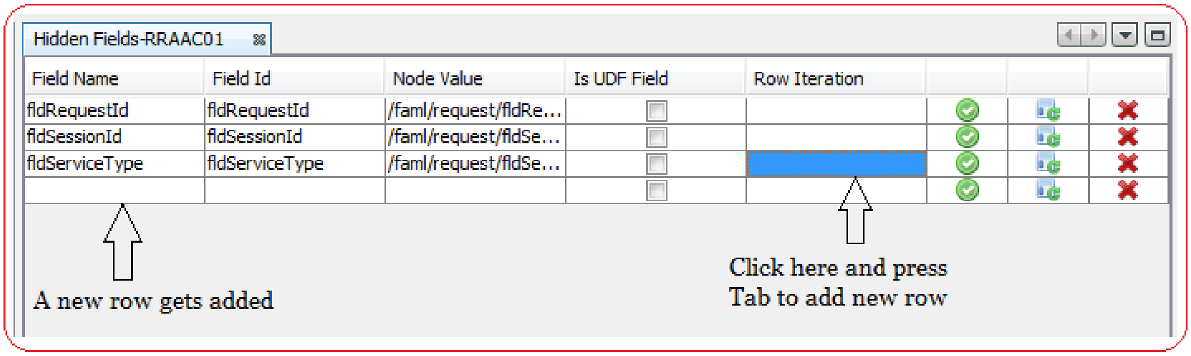
iii. **Node Value:** Node value is the value assigned to this field. It can be either a XPATH or a constant value. For constant value user has to use the syntax string('<value>').

iv. **Is UDF Field:** "isudf" allows a user to pass the field value in UDFDTO array object to the service, if checked, a name value pair of this field can be assigned to UDFDTO by writing following small code snippet in the corresponding service xsl.

```
<xsl:if test="/faml/request/fldudf">
<udfFields>
<xsl:for-each select="/faml/request/fldudf">
<UDFDTO>
<xsl:variable name="udfname"><xsl:value-of select="concat ('/faml/request/',
.)"/></xsl:variable>
<udfName><xsl:value-of select="."/></udfName>
<udfValue><xsl:value-of select="dyn:evaluate($udfname)"/></udfValue>
</UDFDTO>
</xsl:for-each>
</udfFields>
</xsl:if>
```

v. **RowArrayNode:** In certain scenario's, one need to create array of hidden data using an xsl:for-each loop on a particular xpath. To create such an array one can use this column. In this column user needs to enter the xpath on which the iteration has to be done.

To add a new row to this grid table, select the last column of the last row and click tab, a new row will get added, as shown below, where user can define a new hidden field.

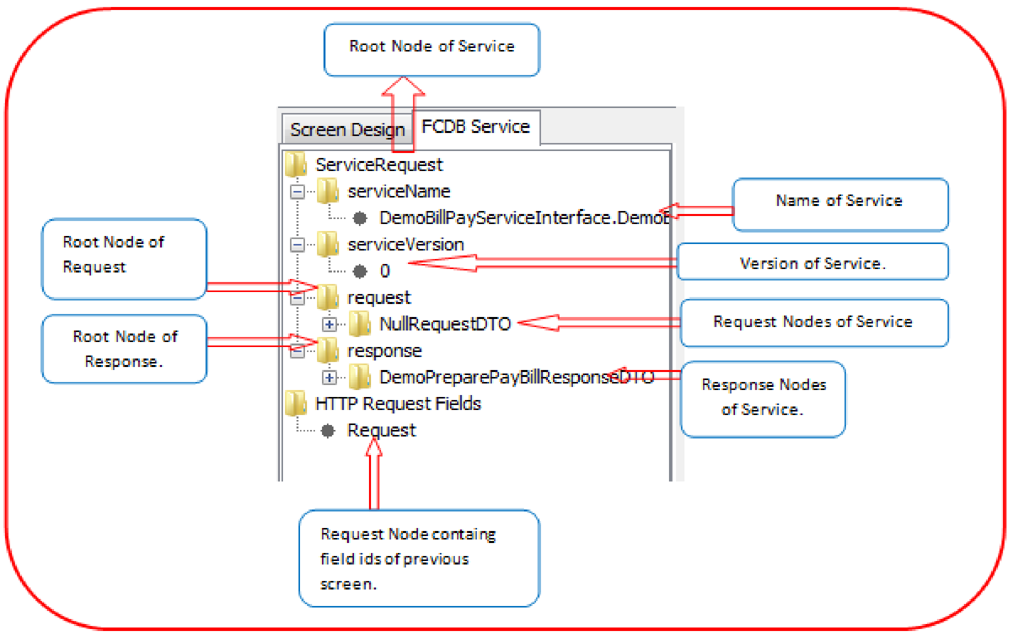


In case of 42 channel or browser based mobile, if a screen has two or more buttons associated with it then there needs to be a distinction between the hidden fields associated with each of these buttons. Hence, the Button Id column is included which takes the value of the button id for each hidden field.

FCDB Service Panel

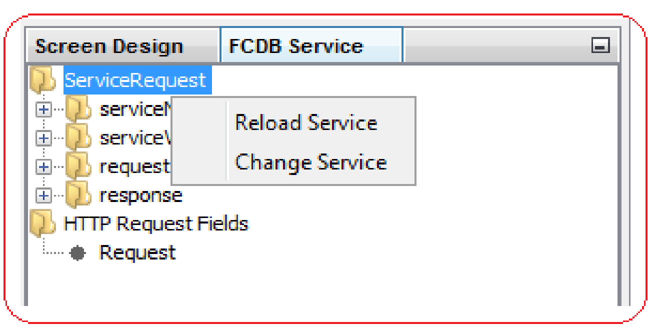
The FCDB Service Tree is shown in "FCDB Service" at the bottom left side of the Channel Workbench Tool on successful loading of a Screen Template Project. The Tree provides a quick view of the service name and version currently invoked by the selected Screen Template Project. The Request DTO needed for the service to execute and the output in the form of Response DTO are also shown in FCDB Service Tree.

This Tree contains the FCDB Service specific details that the user has selected while creating the Screen Template Project. This panel loads the FCDB Service in a tree layout format, where each node represents certain element of the service as shown below,

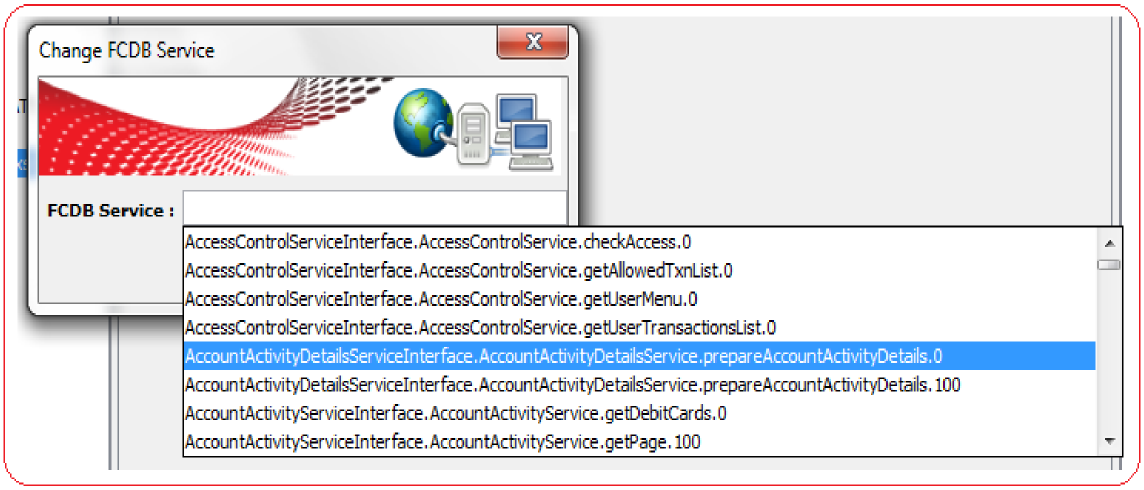


Operations that can be performed on FCDB Service Tree

1. **Changing Service:** User can change the FCDB Service currently being invoked by the Screen Template Project. Service can be changed by right clicking mouse on the root node "ServiceRequest" of FCDB Service tree as shown below,

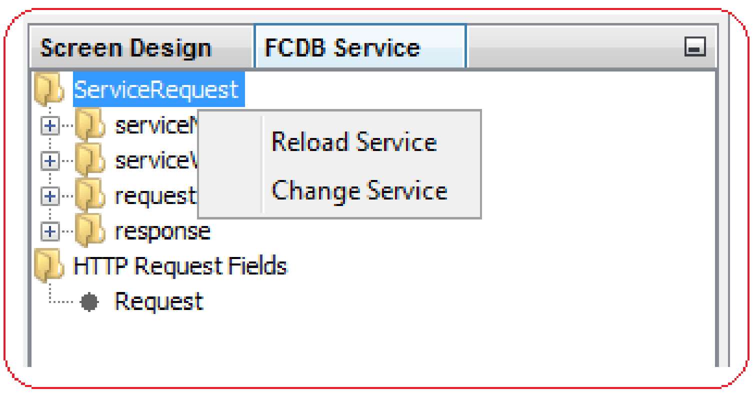


A dialog will appear providing the user with the list of FCDB Services to be selected as shown below,



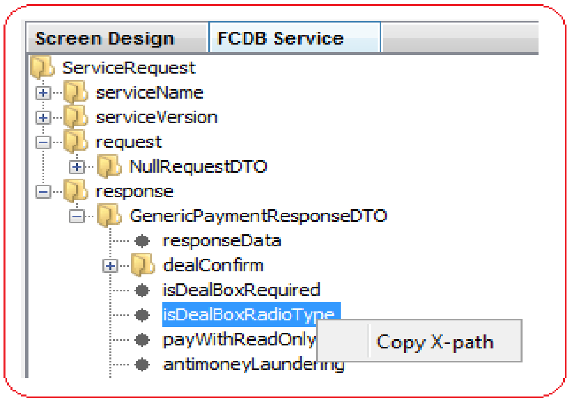
The new Service is loaded into the tool and the tree is refreshed providing the user the details of the new service.

2. **Reloading Service:** The FCDB Service can be reloaded into the tool so that any changes made in the Request DTO or Response DTO gets reflected. Service can be reloaded by right clicking mouse on the root node "ServiceRequest" of FCDB Service tree as shown below,



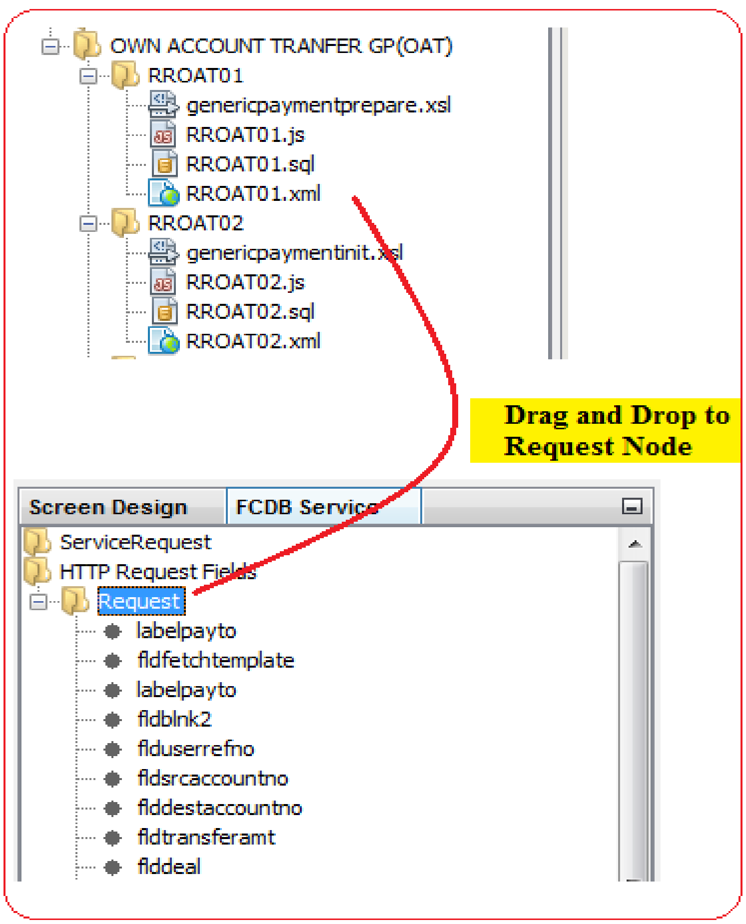
The Service can also be reloaded by double clicking on .xsl file in Screen Template Explorer Tab.

3. **Deriving X-Path:** The FCDB Service Tree can be used to copy X-Path of FCDB Response DTO to a component's node value by simply right clicking mouse on FCDB Service tree node as shown below,



The X-path can also be copied using Drag and Drop.

4. **Field Names of Screen Template Screen:** Field Names of another Screen Template Project can be imported into the currently selected FCDB Service Hierarchy as shown below. User needs to drag the .xml file of another project onto the "Request" node of "HTTP Request Fields". Dropping the design xml will add its fields as child of Request Node, using which a developer will be able to know the request parameter being passed to the current screen.

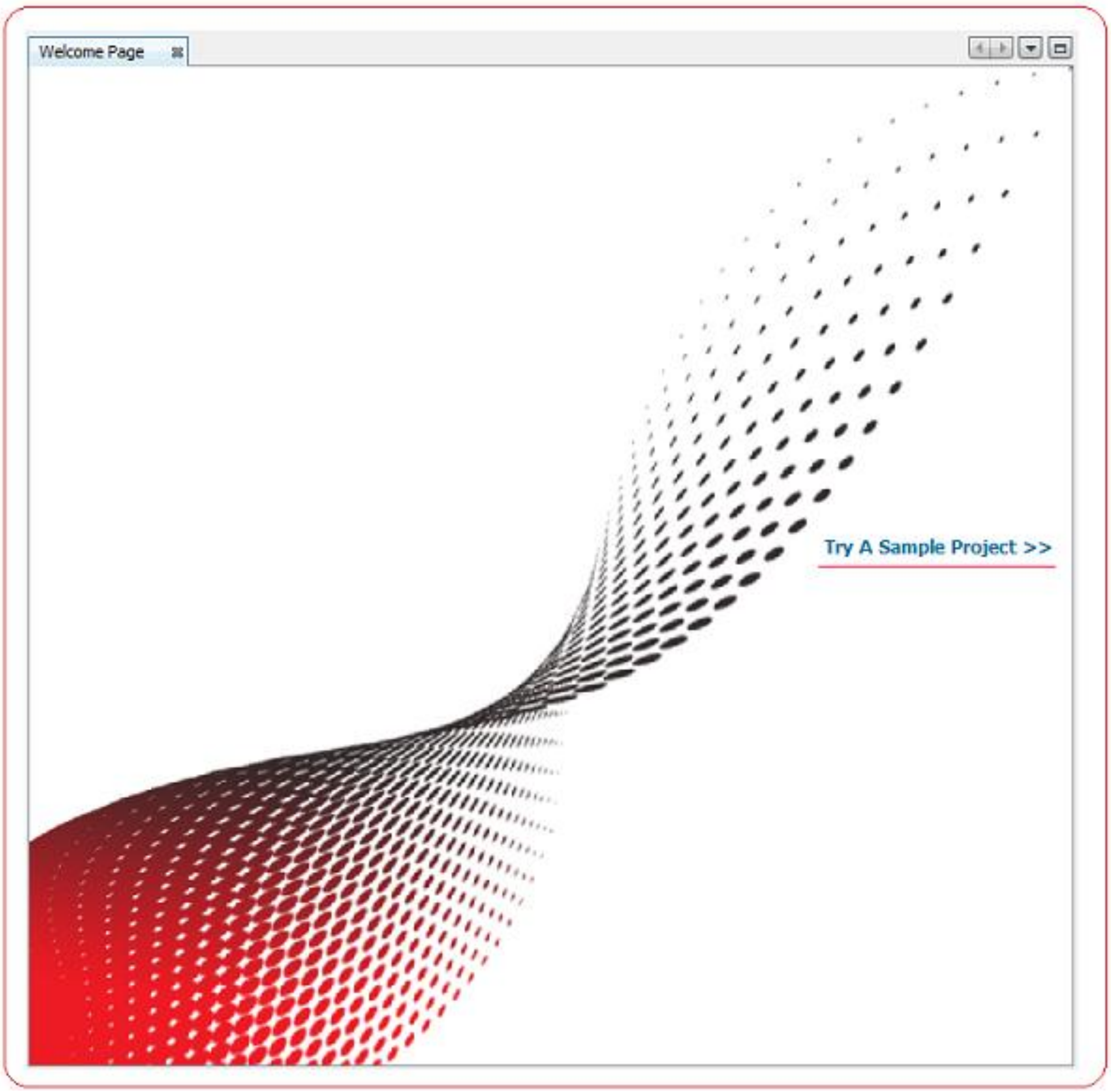


Main Work Area Panel

This panel appears at the center of this tool. This panel consists of several sub-panels which appear depending upon what component of screen design is being currently worked on by the user. Following are the sub-panels which appear inside this panel.

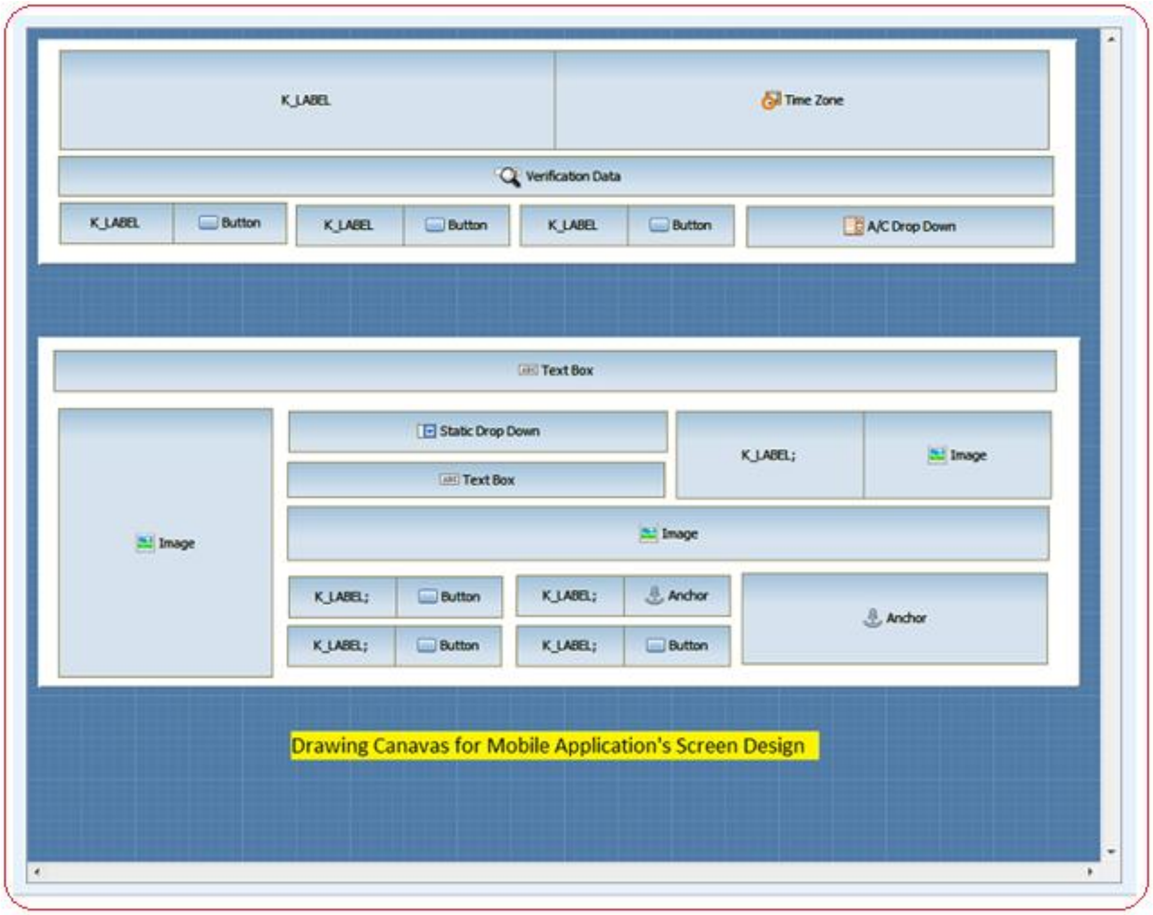
1. Welcome Panel

This is the default panel which will appear in the main work area panel, as shown below, when the tool is launched. Once the user starts working on any screentemplate project, this panel will be replaced by some other sub-panel depending upon what component of screen design user is configuring.



2. Drawing Canvas Panel

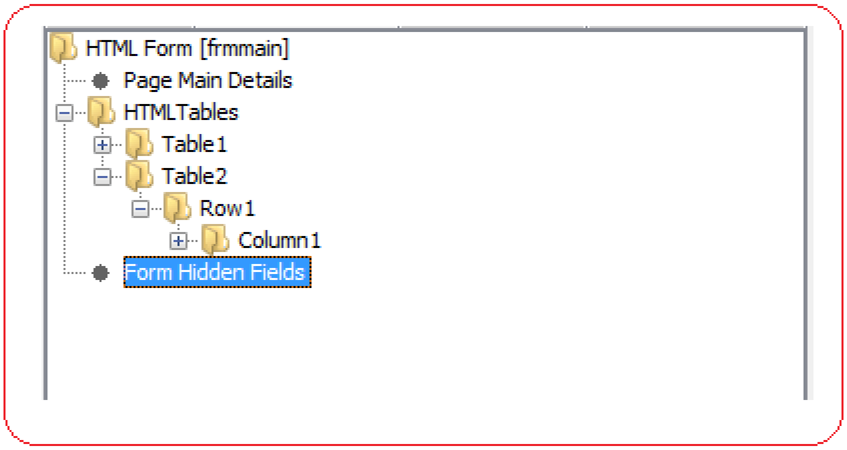
Drawing Canvas Panel for Mobile Apps Screen represents the screen which allows the user to draw table and table components at the required position on the screen as shown below.



Table's and Component's Position X, Position Y, Width and Height will be calculated as % of Screen. Features such as drag/stretch table and components from one position to another and resize of table and components are provided.

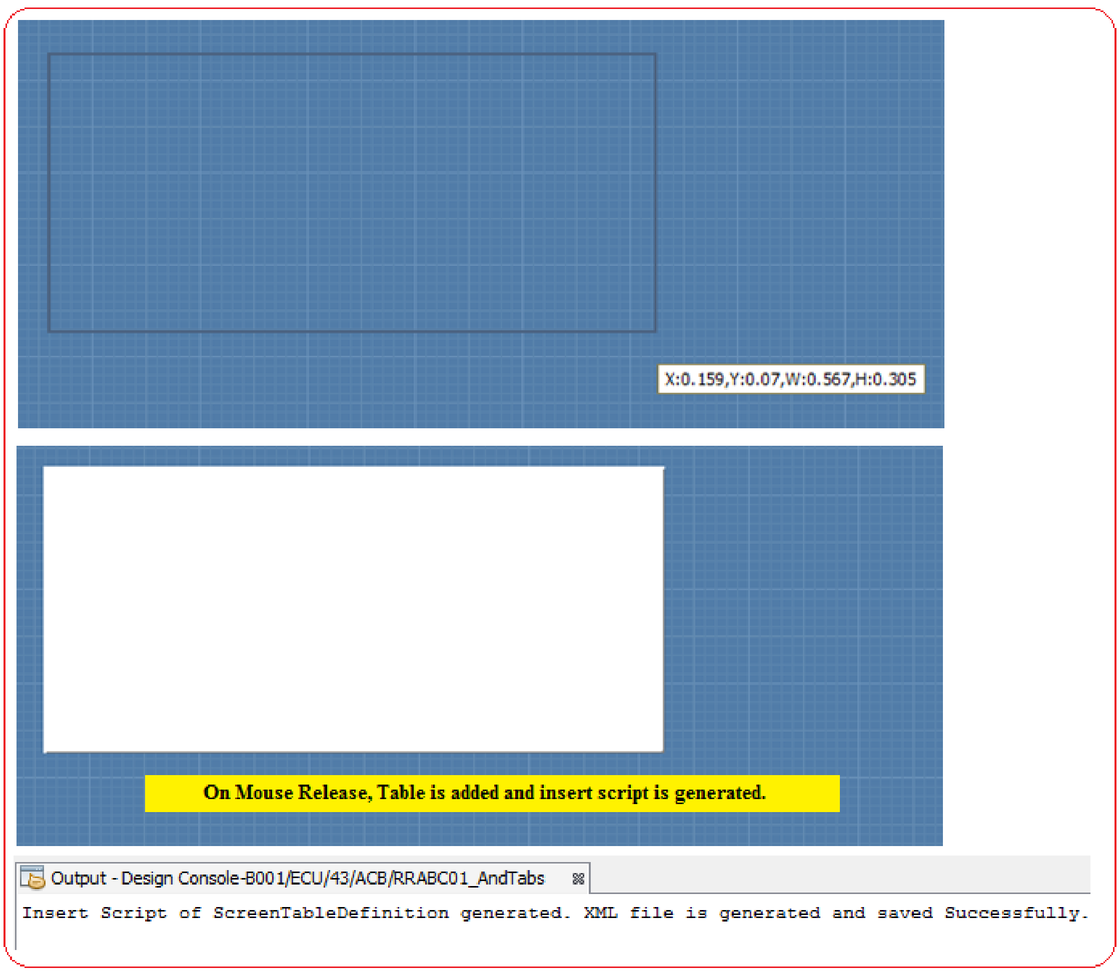
User can drag data types, label types and custom types from html data types pane to table and components on Drawing Panel and their corresponding scripts will be generated and their corresponding DTO's will be updated.

The Canvas only shows tables having display type as 'Default' or 'Wrapper'. In order to view tables with default display type other than 'default' or 'Wrapper' i.e. (Hidden, Modal, Popup), user has to explicitly select the table from the screen layout tree and click on 'Show in canvas option' as shown below:



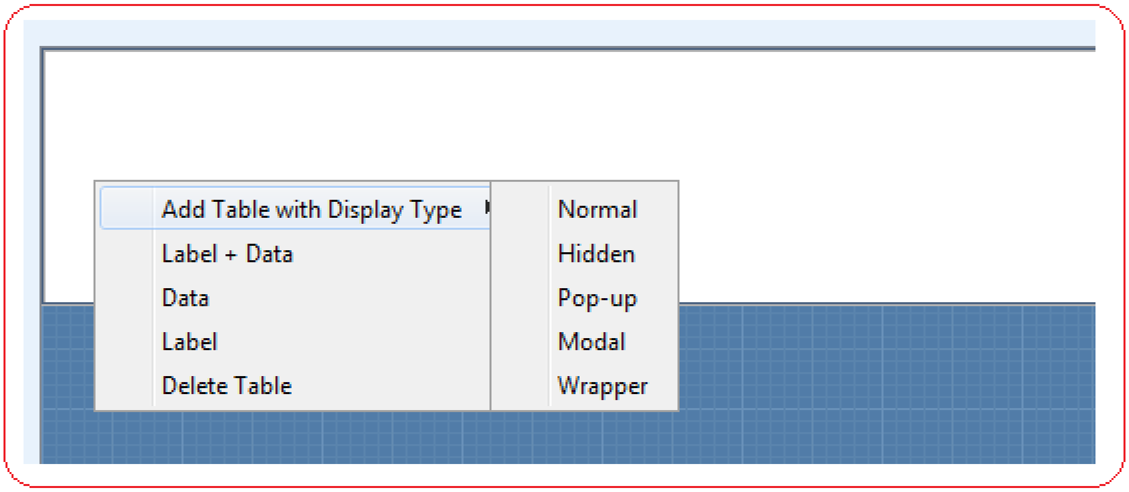
Adding a Table

To add a table in drawing canvas user has to press and drag the mouse at the required position in screen. Table current Relative Position X, Y, Width and Height will be shown while dragging. On releasing the mouse button, table will be added to the canvas and the corresponding scripts will be generated as shown below.



Adding a Component or Table on Top of a Table

To add a component or a table on top of a table to the canvas, user first has to right-click on the table and then click on the type of the component/table to be added. The Component/table with the desired type get added to the corresponding table as shown below.

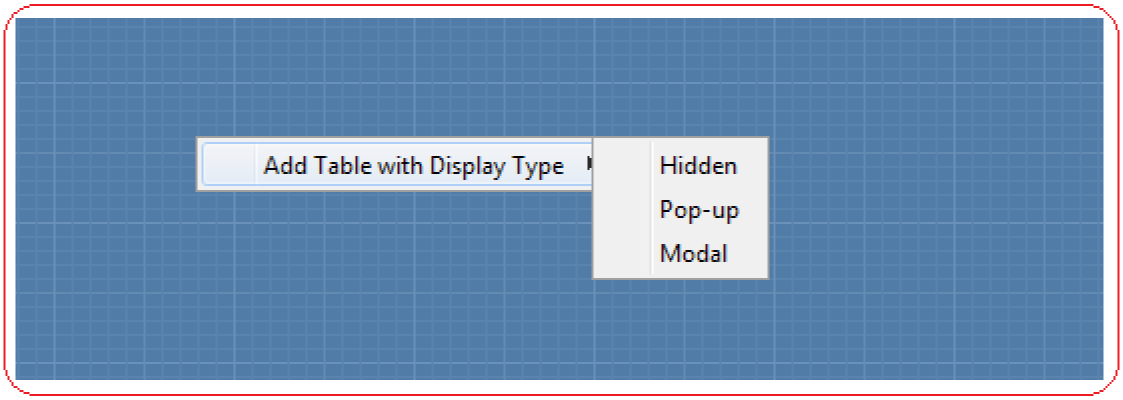


Dragging/Resizing Table/Component

User can also drag/stretch table/Component in order to change the position of the table/Component on screen or resize the table/Component respectively.

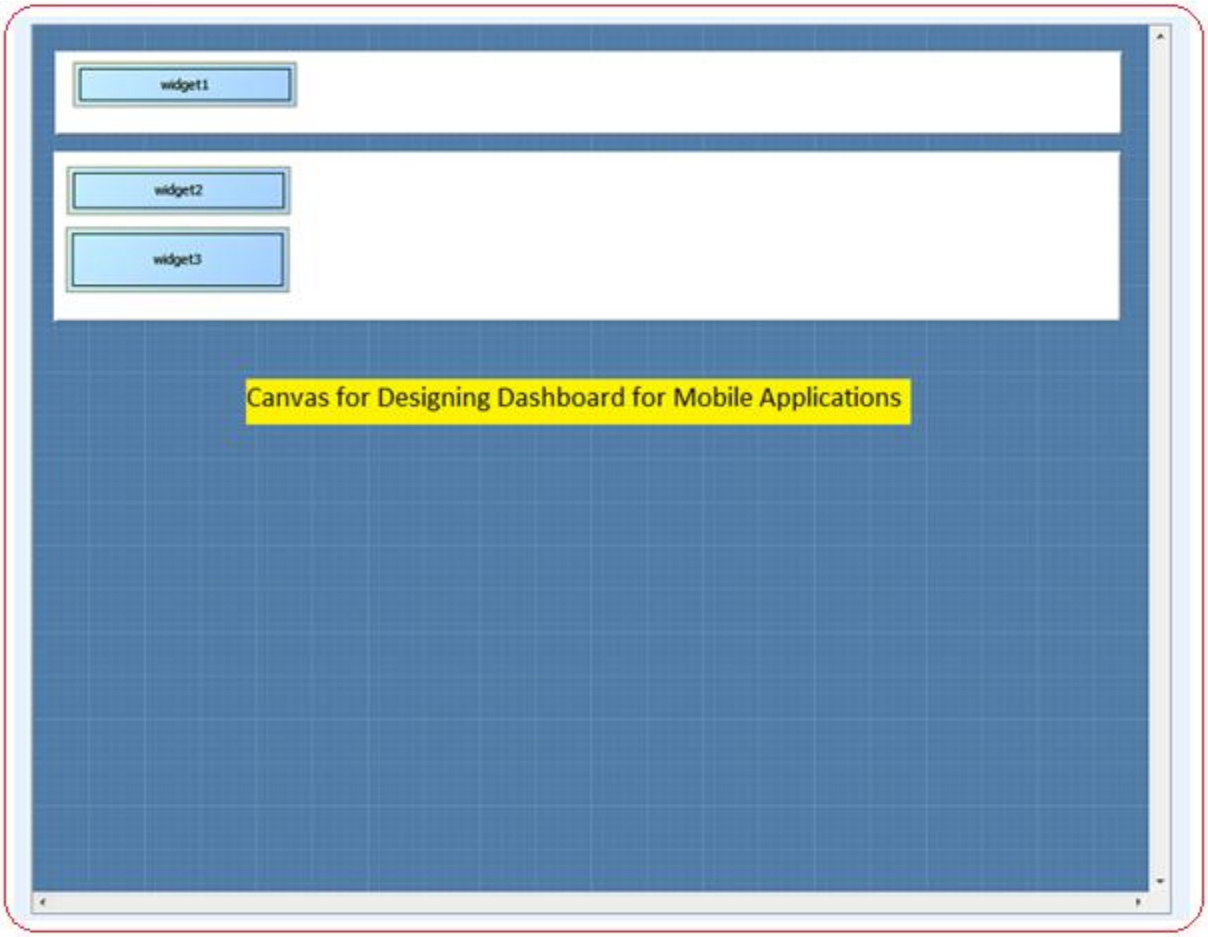
Add a table with display type

User can directly add tables with required display type by right click on the drawing canvas as shown below:



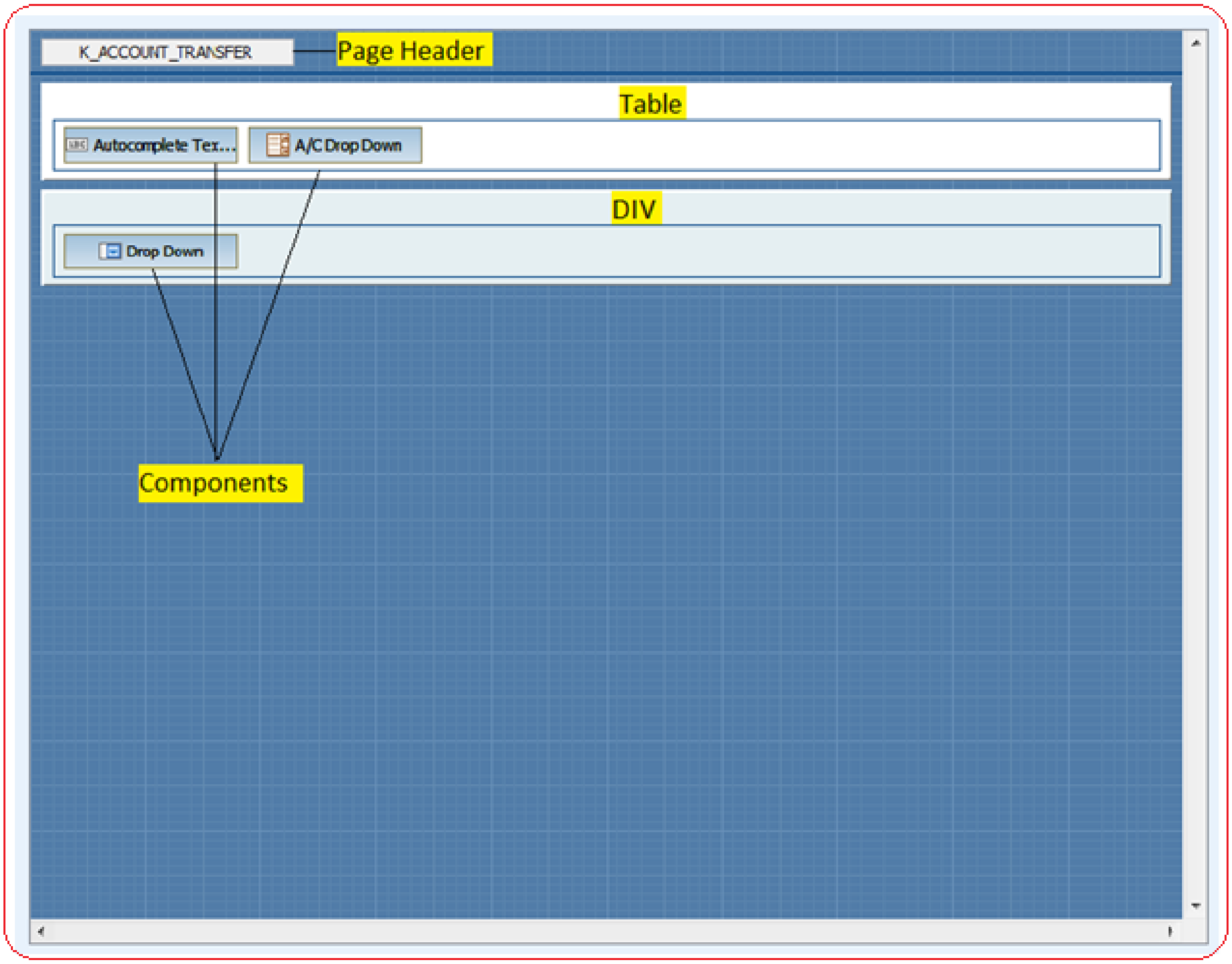
Configuring Dashboard

In case of configuring Dashboard, Drawing Panel represents the screen which allows the user to add widget at the required position on the screen as shown below.


widget1
widget2
widget3
Canvas for Designing Dashboard for Mobile Applications

3. Wireframe Panel

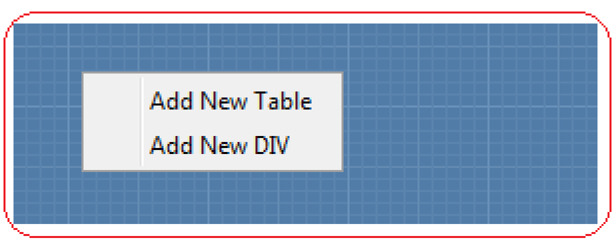
Wireframe Panel provides a wire frame of the tables and components added on the screen for channels other than Mobile Application as shown below.



Creating a screen component can achieved in just few easy step.

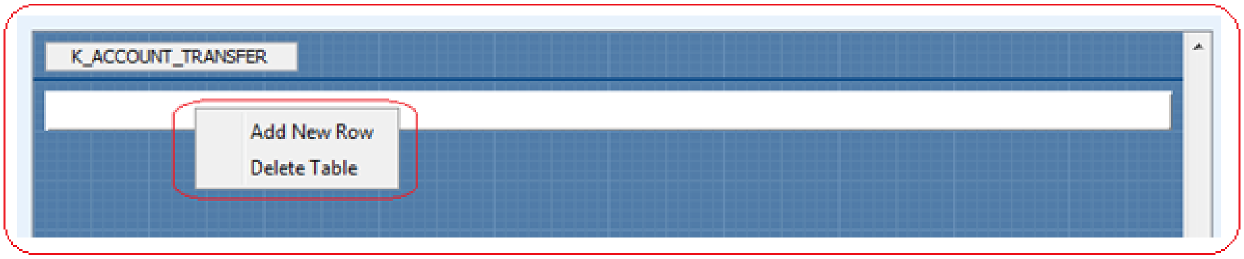
Adding table/DIV

User can directly add a table/DIV from the wire frame panel by clicking on the wire frame panel as shown below.



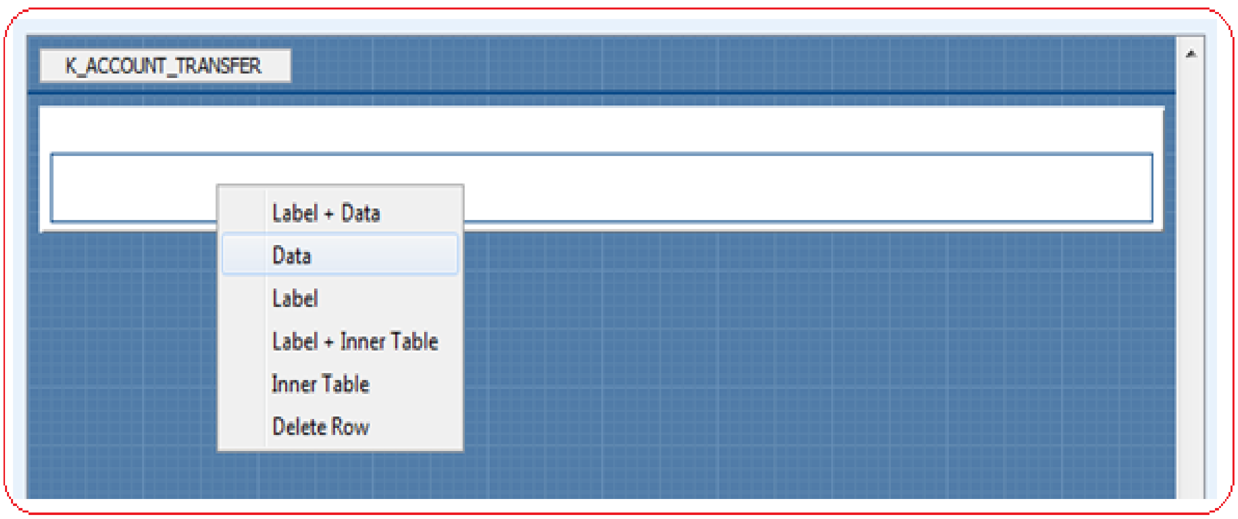
Adding Row

User can directly add a row from the wire frame panel by right clicking on a table/DIV as shown below.



Adding Component

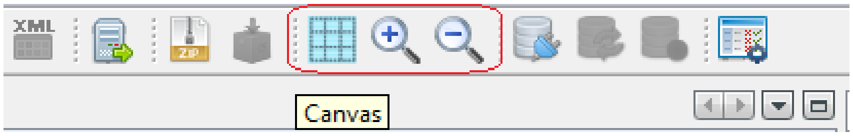
User can also directly add a component from the wire frame panel by right clicking on a row as shown below.



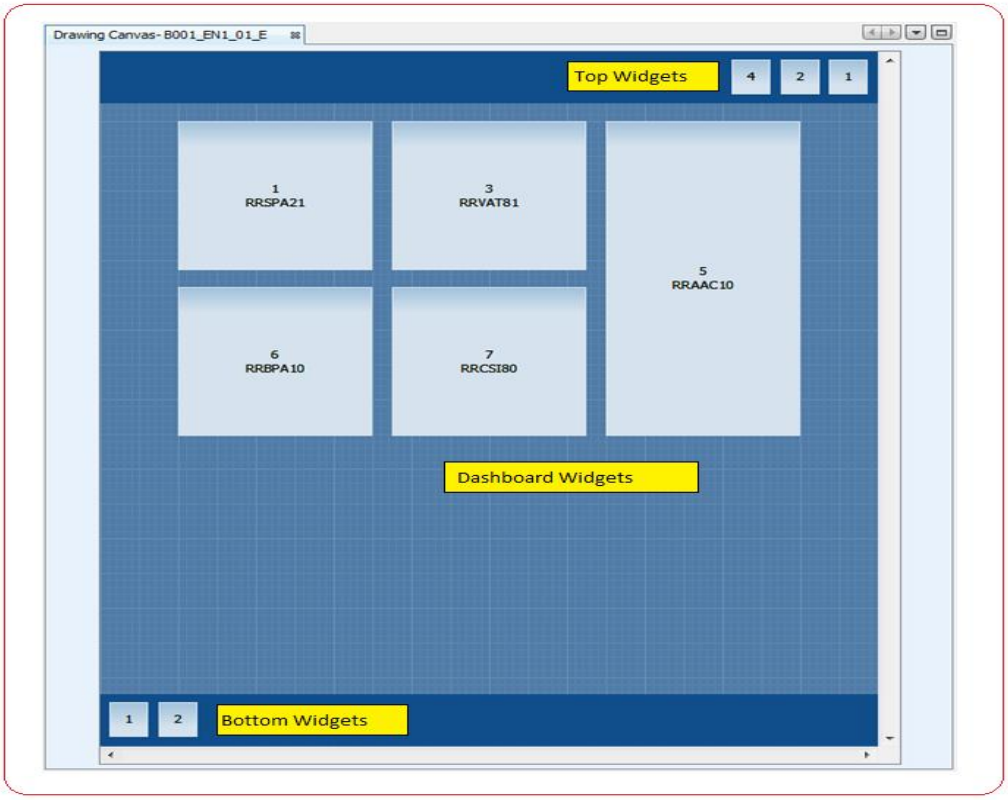
1. Internet Dashboard

Internet Dashboard Wire frame Panel

The Wireframe panel is opened on successful loading of an Internet Dashboard Project on click on an Dashboard project node in Project Explorer Tree in "ScreenTemplate" tab provided at the upper left side of the IDE.
The Wireframe can also be opened by clicking the Show Canvas shortcut icon provided in the tool bar of the IDE. Features to Zoom IN and Zoom OUT from the dashboard canvas are also provided in toolbar as shown below,



Wire frame Panel provides the user a quick view of all the widgets currently added dashboard design as shown below,



Widgets in Dashboard Screen in Oracle Flexcube Direct Banking are placed in three positions.

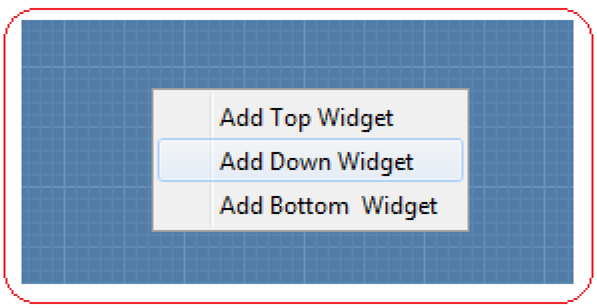
1. **Top Position**: Widgets at this position are placed at the top of the HTML page. Separate space in reserved for them at the top. Widgets here are placed from right to left in increasing order of their sequence number. E.g A widget for user Notifications can be added at top position.

2. **Down Position**: Widgets at this position are placed in middle of HTML page. Positioning of widgets here is based on widgets's sequence number, height scalar and width scalar.

3. **Bottom Position:** Widgets at this position are placed at the bottom of the HTML page. Separate space in reserved for them at the bottom. Widgets here are placed from left to right in increasing order of their sequence number.

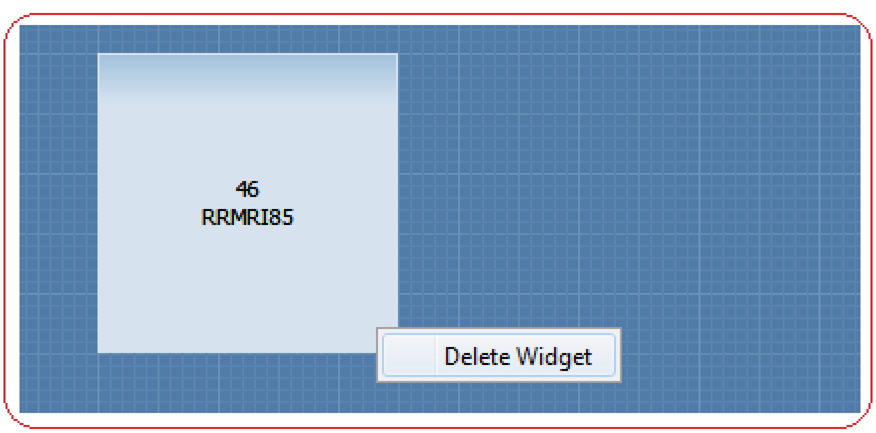
User can perform following operations on the Dashboard wire frame canvas of Internet Channel.

1. **Adding a Widget**
   User can add a widget at a required position from the canvas by right clicking on the wire frame panel as shown below,

   The Widget is added to the dashboard design layout tree in screen design tab in lower left of the IDE. Also the dashboard design xml is saved automatically.

2. **Deleting a Widget**
   User can delete a widget from the canvas by right clicking on a widget as shown below,

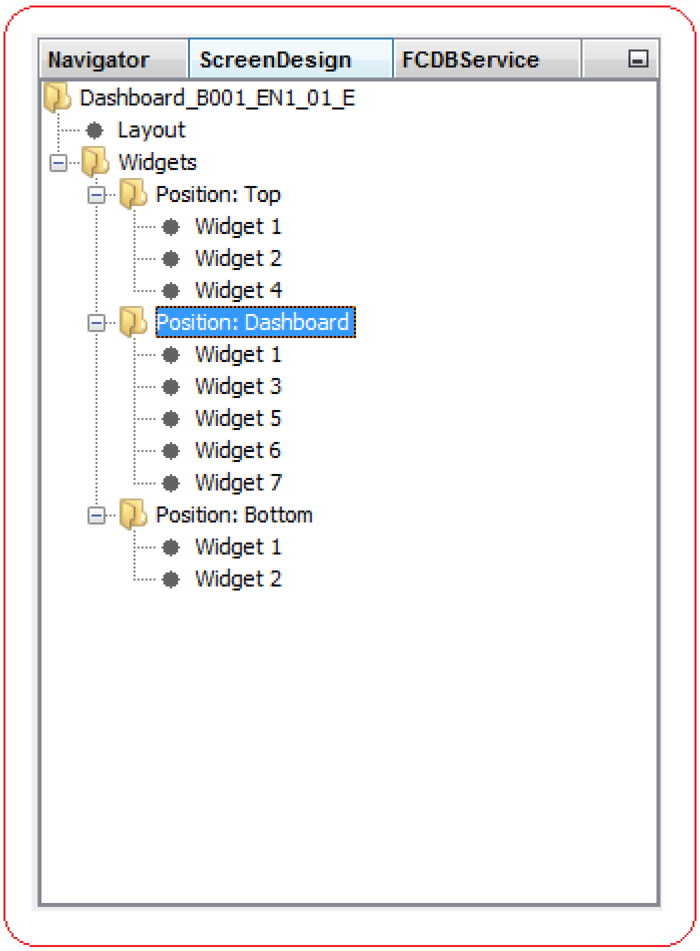
   The Widget is deleted from the dashboard design layout tree in screen design tab in lower left of the IDE. Also the dashboard design xml is saved automatically.

Internet Dashboard Layout Tree

The IDE provides the user a tree based representation of the Internet dashboard design xml. The tree is available to user in "Screen Design" tab at the lower left side of the tool. On successful loading of an Internet Dashboard Project, user needs to click on Dashboard project XML node in Project Explorer Tree in "ScreenTemplate" tab provided at the upper left side of the IDE.

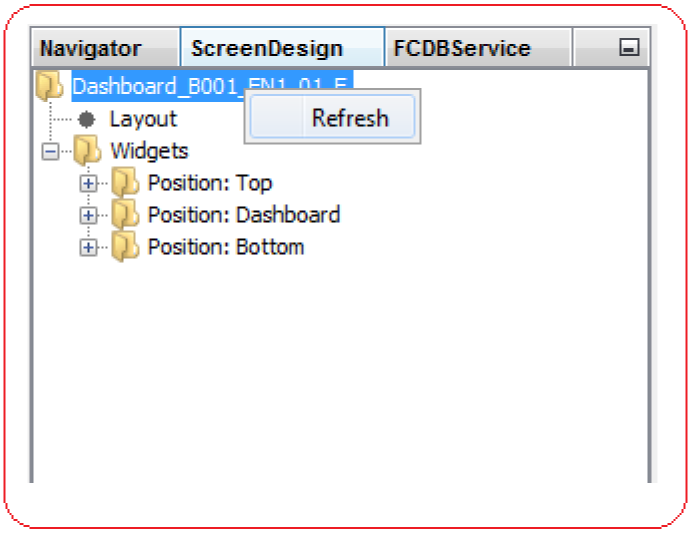
Various operations such as addition of a widget, deletion of a widgets, arranging,re-arranging of widgets can be done using the dashboard layout tree.

Layout tree of a Dashboard layout project is shown below,



Description of Nodes

1. **Root Node:**
   The Root Node contains the description of the dashboard project that is currently selected. It takes the format "Dashboard_Entity ID_User Type_Channel ID_Dashboard ID".
   A popup is shown on right click of this node as shown below,



   Click on menu item **Refresh** to validate the dashboard design xml. Any errors or warnings due to an invalid entries in the design xml is displayed in output console panel at the bottom of the IDE.

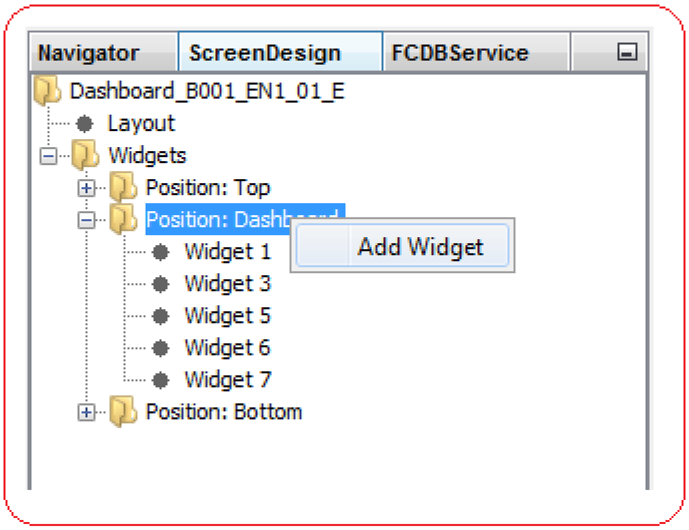
2. **Layout Node:**
   A single click on this node opens a palette for the user to edit the layout properties of dashboard at the right hand side of IDE. For more information of various properties of the layout please refer layout properties section.

3. **Widgets Node:**
   The Widgets node consists of the defined positions where the widgets are being placed on Dashboard. Widgets added at a position are placed at that defined position on Dashboard.
   The positions currently defined are as below,
   - ✓ **Top Position:** Widgets at this position are placed at the top of the HTML page. Separate space is reserved for them at the top. Widgets here are placed from right to left in increasing order of their sequence number.
   - ✓ **Dashboard Position:** Widgets at this position are placed in middle of HTML page. Positioning of widgets here is based on widgets's sequence number, height scalar and width scalar.
   - ✓ **Bottom Position:** Widgets at this position are placed at the bottom of the HTML page. Separate space in reserved for them at the bottom. Widgets here are placed from left to right in increasing order of their sequence number.

   A single click on any widget node opens a pallete for the user to edit the widget properties at the right hand side of IDE. For more information of various properties of the layout please refer widget details section.

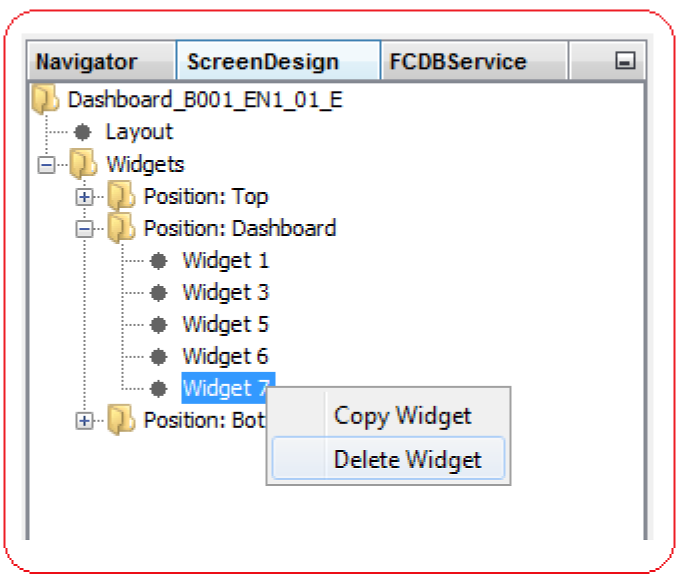
User can perform following operations on the Dashboard Layout Tree.

1. **Adding a Widget**
   User can add a widget by right clicking on a position node where the widget is to be added as shown below,



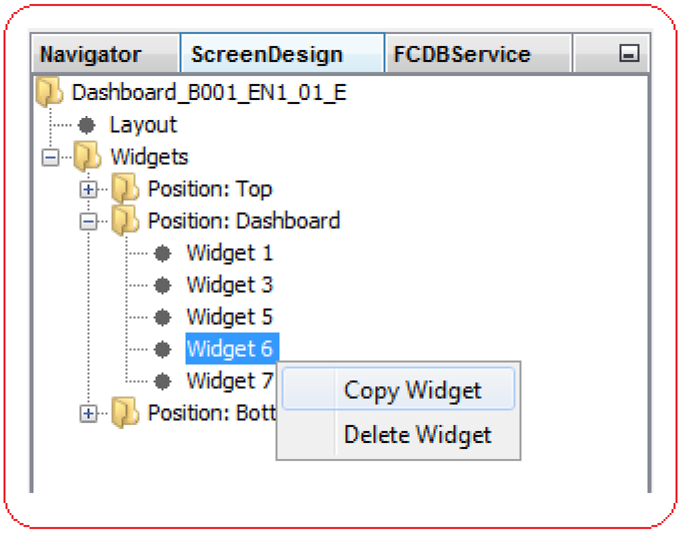
   The Widget is added to the dashboard design layout tree. Also the dashboard design xml is saved automatically.

2. **Deleting a Widget**
   User can delete a widget by right clicking on a widget as shown below.

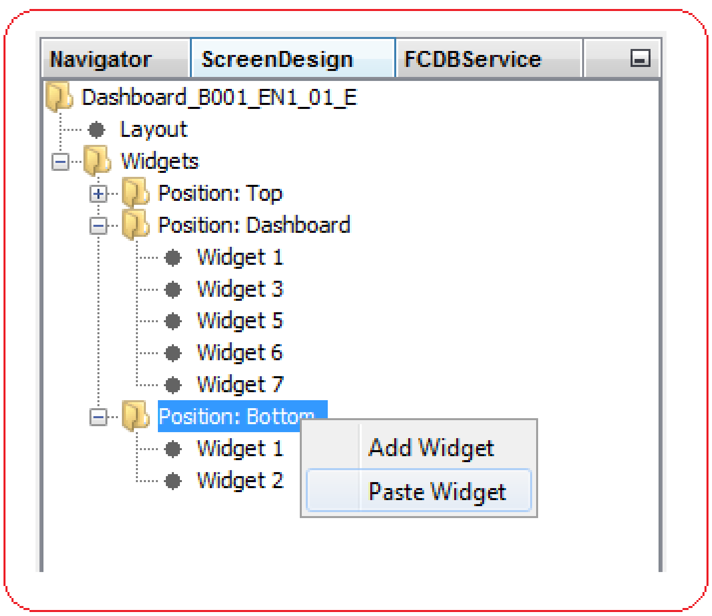


   The Widget is deleted from the dashboard design layout tree. Also the dashboard design xml is saved automatically.

3. **Copying a Widget**
User can copy a widget across dashboard projects by right clicking on a widget and selecting "Copy Widget" option as shown below.



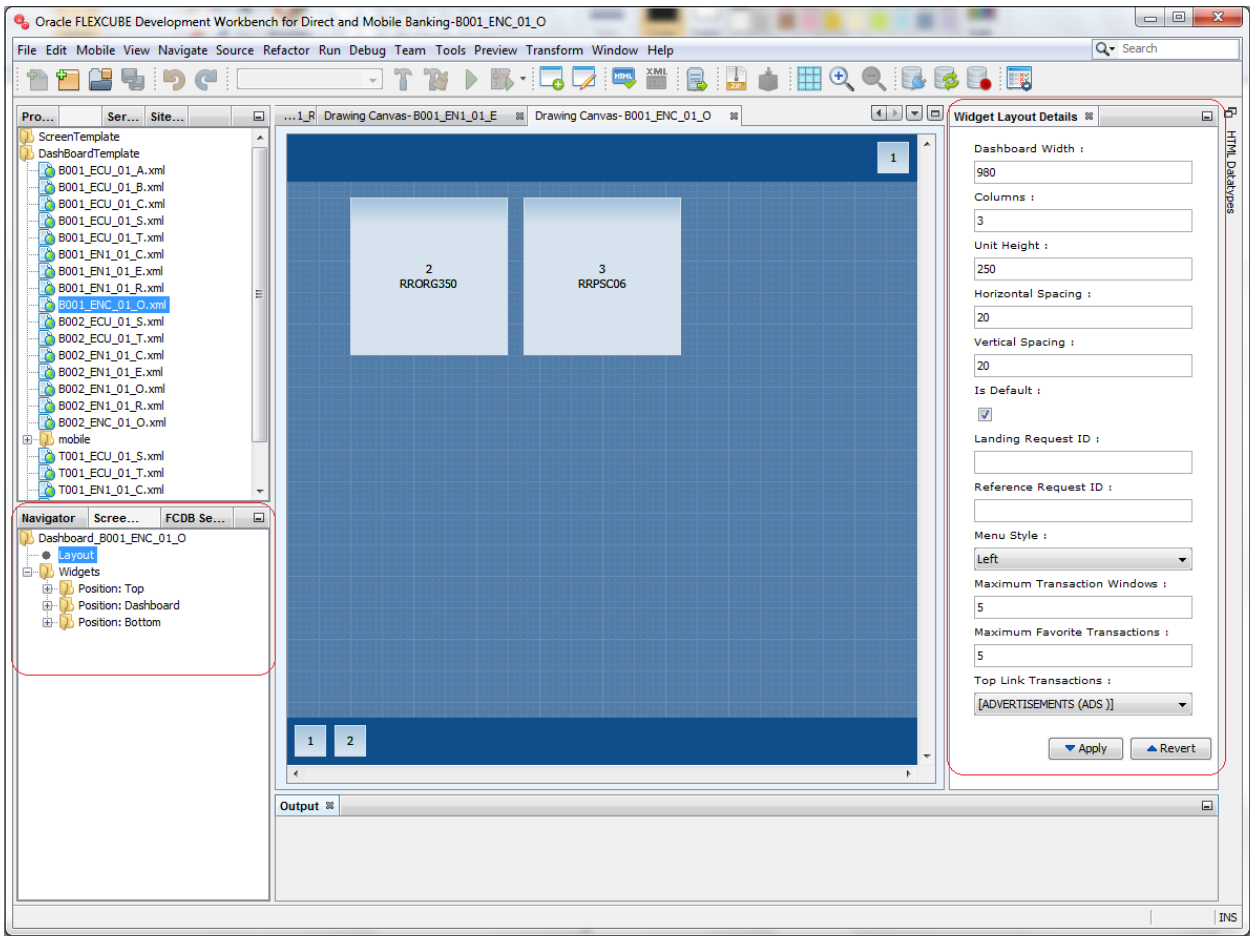
Right Click on a widget position node to paste the copied widget as shown below,



4. **Arranging/Re-arranging Widgets**
Drag and Drop functionality is implemented in Dashboard layout tree for users to change the sequencing and positioning of widgets according to thier needs.
User can drag a widget from a position and drop it at another position node to change the position of the widget. The Sequence number will be automatically generated according to the new position where the widget is dropped.

Widget Layout Details

Widget Layout Details appears as a pallete to users to edit layout properties of Dashboard Design Template.
The Layout pallete is opened at the right hand side of IDE on clicking on Layout node of Screen Design Layout panel, as shown below,



Widget Layout Details Panel helps user to configure various layout properties of the dashboard design.

Description of Fields:

- ✓ **Dashboard Width:** Dashboard Width represents the dashboard display area resolution in pixels. Numeric Values greater than 960 are allowed in this field. For Oracle FCDB application, the ideal configuration for dashboard width is 1024 pixels. By default, value 1024 pixels is set in this field. It is recommended for user to enter proper resolution size in this field. E.g. 1024, 1152, 1280, etc. for proper alignments in dashboard

- ✓ **Columns:** This field represents the number of horizontal widgets supported in single row of dashboard. Number of columns should be greater than or equal to 1. By default the columns field is set to 3.

- ✓ **Unit Height:** This field represents the fix unit height of widgets in pixels for a dashboard. The Actual height of a widget is calculated by multiplying unit height with Height Scalar of a widget found in Widget Details Configuration pallete. By Default, unit height is set to 250 pixels.

- ✓ **Horizontal Spacing:** This field represents the horizontal spacing in pixels between two widgets horizontally. Only Numeric Values are allowed in this field.

- ✓ **Vertical Spacing:** This field represents the Vertical spacing in pixels between two widgets vertically. Only Numeric Values are allowed in this field.

- ✓ **Is Default:** This field represents a flag to indicate default dashboard for a entity_usertype_channel combination. The flag will be used in admin transactions to select default dashboard style while creating and modifying user.

- ✓ **Landing Request:** This field represents the Oracle FCDB request identifier. This field is mandatory if simple dashboard flag is true.

- ✓ **Reference Request ID:** Using this field, widgets can be grouped together to show them on transaction screen. Value of this field should be a request id of transaction screen.

- ✓ **Menu Style:** This field is used to set the style of menu. Possible styles for menu are Top, Left and Bottom.

- ✓ **Maximum Transaction Windows:** This field is used to set the maximum number of transaction windows that can be opened simultaneously. Only Numeric values greater than or equal to 1 are allowed. Default Value for Maximum Transaction that can opened is 5.

- ✓ **Maximum Favorite Transactions:** This field is used to set the maximum number of transactions that can be marked as Favorite by user. Only Numeric Values are allowed in this field. Default value is 5.

- ✓ **Quick Link Transactions:** This field indicates list of transactions that should be displayed as Quick Links at the Top of the Dashboard.

Description of Buttons

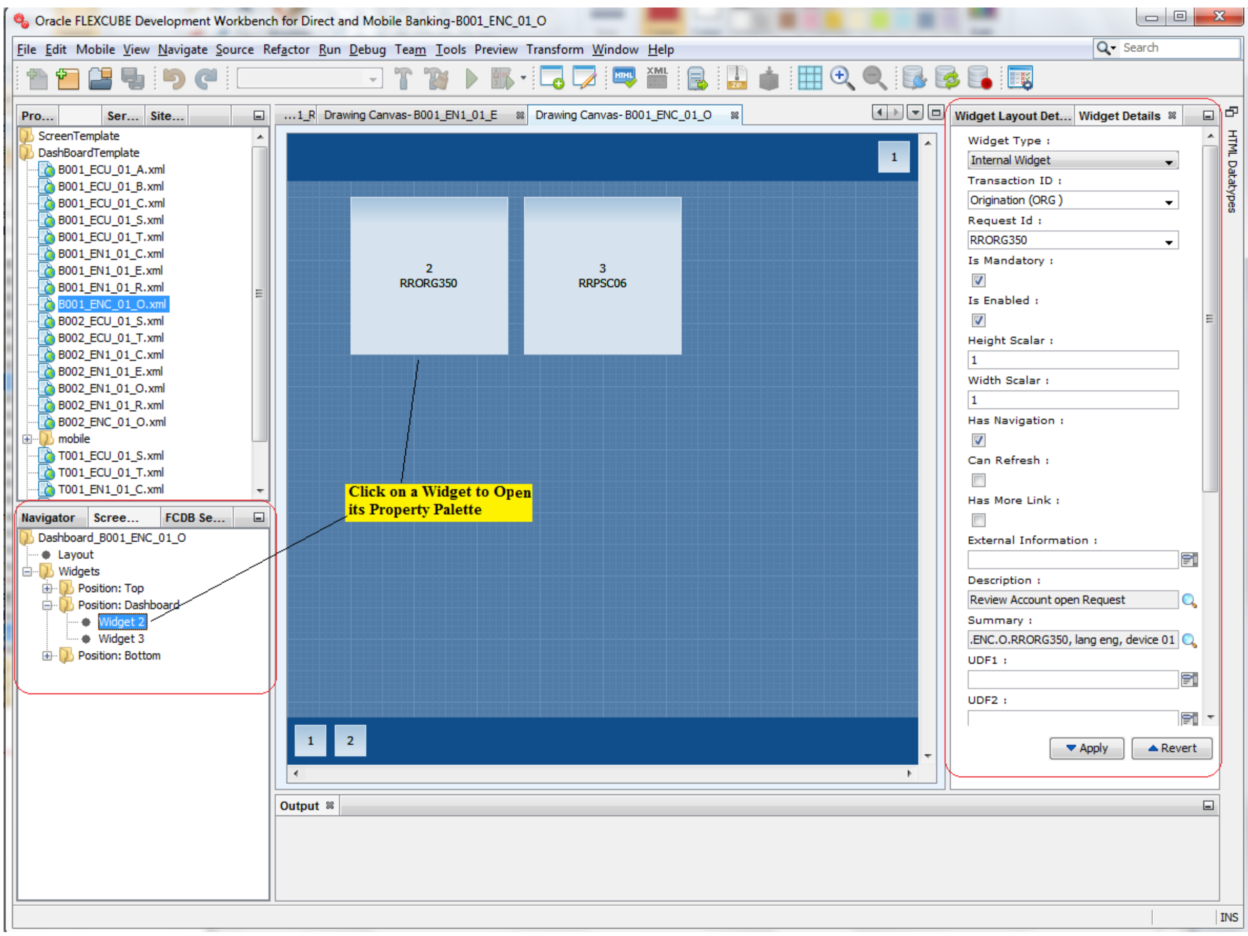
1. 

: Once the user has entered the above details, they need to click on "Apply" button to apply the changes done. If not clicked the changes will not get reflected on revisiting this panel.
2. 

: If the user has not clicked on "Apply" and wants to go back to last applied changes, they can click on "Revert" button.

Widget Details

Widget Details appears as a pallete to users to edit widget properties of Dashboard Design Template.
The properties pallete is opened at the right hand side of IDE on clicking on any Widget node of Screen Design Layout panel or any widget from the wireframe canvas as shown below,



Widget Details Panel helps user to configure various properties of widgets of the dashboard design.

Description of Fields:

- ✓ **Transaction ID:** This field represents Transaction identifier which is shown in this widget. It is mandatory field for an Internal Widget.

- ✓ **Request ID:** This field represents landing screen identifier which is shown in this widget. It is mandatory field for an Internal Widget.

- ✓ **Widget Type:**This field indicates whether widget is an Internal, External or Script Widget. Transaction ID and Request ID are mandatory for an Internal Widget. External Information is mandatory for External or Script Widget.

- ✓ **Widget Position:** This field represents the positions that are defined to place a widget on dashboard in Oracle FCDB application. Following positions are defined in application,
  - **Top Position**: Widgets at this position are placed at the top of the HTML page. Separate space in reserved for them at the top. Widgets here are placed from right to left in increasing order of their sequence number. E.g A widget for user Notifications can be added at top position.
  - **Down Position**: Widgets at this position are placed in middle of HTML page. Positioning of widgets here is based on widgets's sequence number, height scalar and width scalar.
  - **Bottom Position**: Widgets at this position are placed at the bottom of the HTML page. Separate space in reserved for them at the bottom. Widgets here are placed from left to right in increasing order of their sequence number.

- ✓ **Widget Sequence**: A widget sequence is unique for a position defined for dashboard in Oracle FCDB application. Widgets are placed in increasing order of their sequence at a postion on dashbaord.

- ✓ **Is Mandatory**:This field represents flag to indicate whether this widget is mandatory on dashboard or not.

- ✓ **Is Enabled**:This field represents flag to indicate whether this widget is to be considered in application or not.

- ✓ **Height Scalar:** This field indicates the scalar value (no of units) using which widget height will get applied on screen. By default, the height scalar is set to 1.

- ✓ **Width Scalar**: This field indicates the scalar value (no of units) using which widget width will get applied on screen. By default, the width scalar is set to 1.

- ✓ **Has Navigation:** This field represents a flag to indicate whether the widget requires title bar or not on dashboard. If this flag is not set, then "Can Refresh" and "Can Minimize" flag are not considered and then hence are disabled.

- ✓ **Can Refresh:** This field represents a flag to indicate whether refresh functionally is to be provided or not on widget. This field is enabled only if "Has Navigation" checkbox is checked.

- ✓ **Has More Link**: This field represents a flag to indicate whether parent transaction link is to be provided or not on widget. This field is enabled only if "Has Navigation" checkbox is checked.

- ✓ **External Information**: Set any external information required for dashboard configuration. For e.g. HTML script can be entered in this field like <script src=""http://www.gmodules.com/ig/ifr?url=http://www.pixelmedia.nl/gmodules/ucc.xml;;;up_fro

mcur=INR;;;up_tocur=USD;;;synd=open;;;w=250;;;h=130;;;title=;;;lang=en;;;country=IN;;;border=%23ffffff%7C3px%2C1px+solid+%23999999;;;output=js""></script>

- ✓ **Description**: This field not available to user to edit. It is automatically derived by tool. The field provides description of the widget. If "Missing Data Map Entry.." is shown in this field, then the user needs to configure the label using the lookup icon provided besides the field.
- ✓ **Summary**: This field not available to user to edit. It is automatically derived by tool. The field provides summary of the widget. If "Missing Data Map Entry.." is shown in this field, then the user needs to configure the label using the lookup icon provided besides the field.
- ✓ **UDF1**: This field is used for bank purpose.
- ✓ **UDF2**: This field is used for bank purpose.
- ✓ **UDF3**: This field is used for bank purpose.
- ✓ **UDF4**: This field is used for bank purpose.
- ✓ **UDF5**: This field is used for bank purpose.
- ✓ **Has Button:** This field represents a flag to indicate whether button for some functionality is to be provided in the title bar of the widget.
- ✓ **Button Text:** This field represents the text to be displayed as label on top of button. The Label can be configured using the lookup icon provided besides this field. The keyword should start with K_.
- ✓ **Button Function:** This field represents the function to be called on click of button.
- ✓ **Launcher Class**: This field is associated with top widgets.
- ✓ **Content Class**: This field is associated with Working area. Using this field, widget background image can be set.

Description of Buttons

1. 

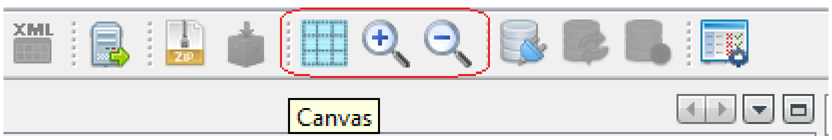
: Once the user has entered the above details, they need to click on "Apply" button to apply the changes done. If not clicked the changes will not get reflected on revisiting this panel.
2. 

: If the user has not clicked on "Apply" and wants to go back to last applied changes, they can click on "Revert" button.

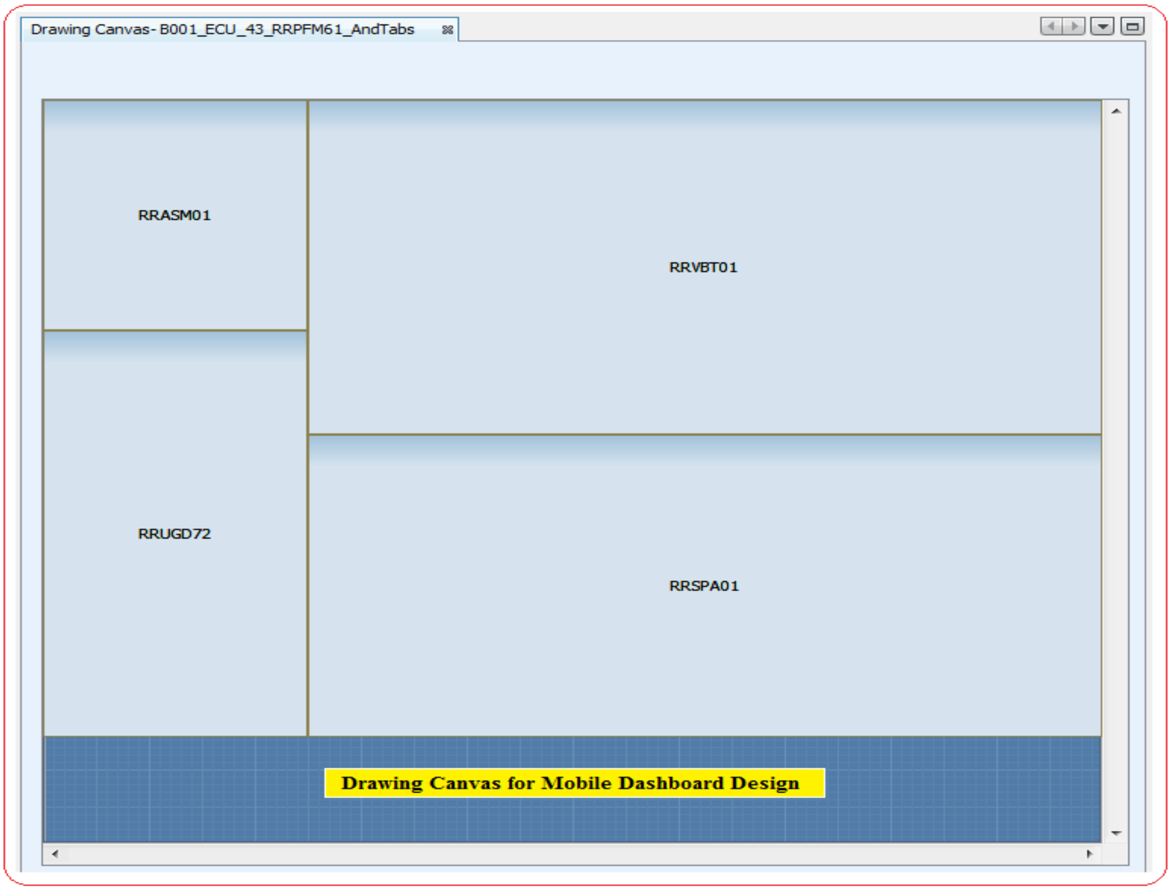
2. Mobile Dashboard

Drawing Canvas for Mobile Dashboard Design

The Drawing Canvas for designing Mobile Dashboard is opened on successful loading of an Mobile Dashboard Project on click on an Mobile Dashboard project node in Project Explorer Tree in "DashboardTemplate" tab provided at the upper left side of the IDE.
The Canvas can also be opened by clicking the Show Canvas shortcut icon provided in the tool bar of the IDE. Features to Zoom IN and Zoom OUT from the dashboard canvas are also provided in toolbar as shown below,

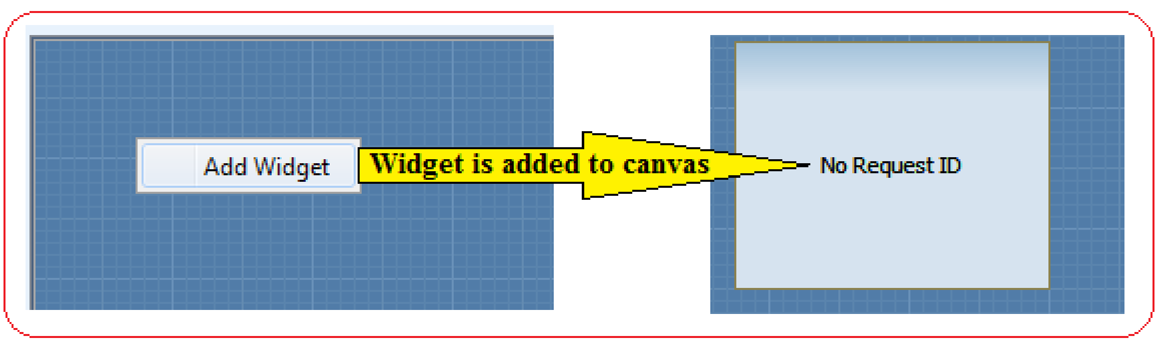


The Canvas allows the user to design screen for Mobile Dashboard. The Canvas represents the mobile screen which allows the user to draw widgets at the required position on the screen as shown below.



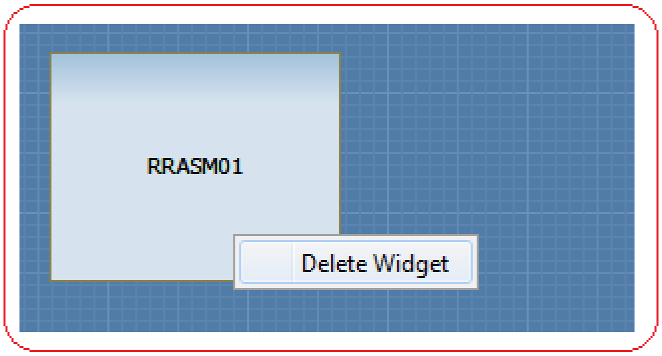
User can perform following operations on the Dashboard wire frame canvas of Internet Channel.

1. **Adding a Widget**
   User can add a widget at a required position from the canvas by right clicking on the canvas as shown below,



A new Widget is added to the dashboard design layout tree in screen design tab in lower left of the IDE. The Request ID is not set for the new widget added. Also the dashboard design xml is saved automatically.

2. **Deleting a Widget**
   User can delete a widget from the canvas by right clicking on a widget as shown below.



   The Widget is deleted from the dashboard design layout tree in screen design tab in lower left of the IDE. Also the dashboard design xml is saved automatically.

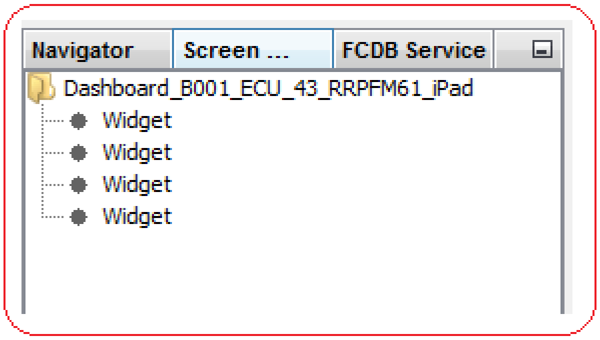
3. **Dragging/Resizing Widget:** User can also drag/stretch widget in order to change the position of the widget on screen or resize the widget respectively. Feature to drag and drop a widget is provided.

Mobile Dashboard Layout Tree

The IDE provides the user a tree based representation of the Mobile dashboard design xml. The tree is available to user in "Screen Design" tab at the lower left side of the tool on successful loading of an Mobile Dashboard Project. To load a dashboard project into workspace, user needs to click on Dashboard project XML node in Project Explorer Tree in "ScreenTemplate" tab provided at the upper left side of the IDE.

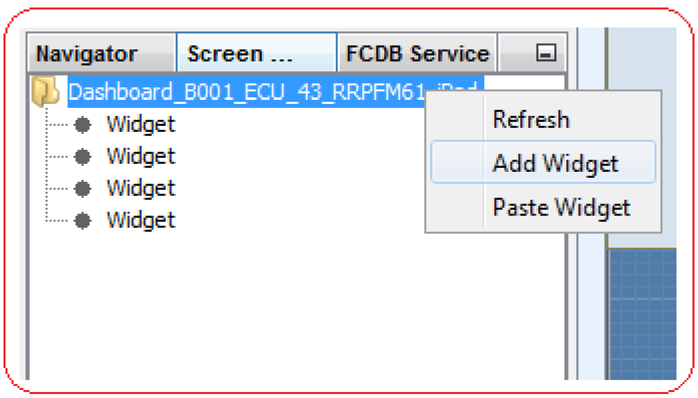
Various operations such as adding a widget, deleting a widget, copying a widget can be done using the dashboard layout tree.

Layout tree of a Mobile Dashboard project is shown below,



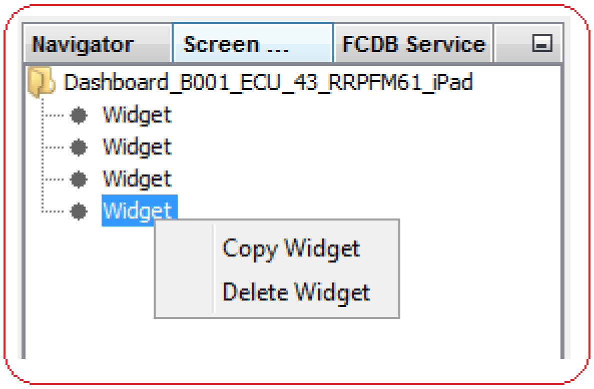
Description of Nodes

1. **Root Node:**
   The Root Node contains the description of the dashboard project that is currently selected. It takes the format "Dashboard_Entity ID_User Type_Channel ID_Dashboard ID_User Agent". A popup to add/paste widget is shown on right click of this node as shown below,



- ✓ **Add Widget**: The Widget is added to the dashboard design layout tree. Also the dashboard design xml is saved automatically.
- ✓ **Paste Widget**: The Paste Widget Menu Item is shown only if a widget is copied from the Layout Tree. On click of this menu item, a widget is added with the same properties as the source widget. Also the dashboard design xml is saved automatically.

2. **Widgets Node**: A popup to delete/copy a widget is shown on right click as shown below,



The user can perform the following operation on a widget.

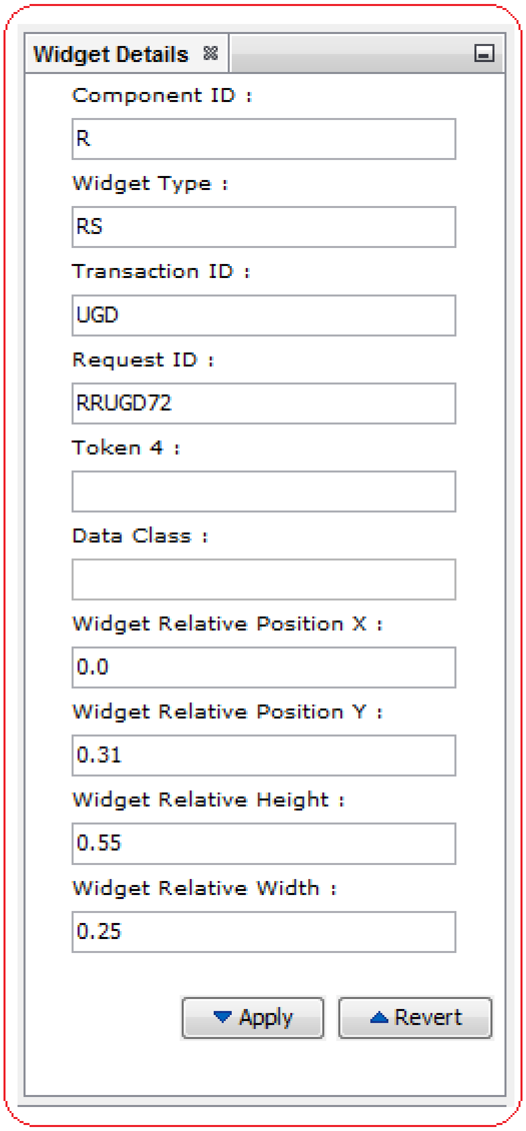
- ✓ **Delete Widget**: The Widget is removed from the dashboard design layout tree. Also the dashboard design xml is saved automatically.

- ✓ **Copy Widget**: Click on menu item to copy a widget. The widget can be pasted by right clicking on the root node of the dashboard layout tree.

A single click on any widget node opens a pallete for the user to edit the widget properties at the right hand side of IDE. For more information of various properties of the layout please refer widget details section below.

Widget Details

Widget Details appears as a pallete to users to edit properties of widget of Dashboard Design Template.
The properties pallete is opened at the right hand side of IDE on clicking on any Widget node of Screen Design Layout panel or any widget from the canvas as shown below,



Widget Details Panel helps user to configure various properties of widgets of the dashboard design.

Description of Fields.

1. **Component ID:** Component ID to be used for widget.

2. **Widget Type**: Type of Widget. Widgets are rendered and painted on screen according to its type.

3. **Transaction ID:** This field represents Transaction identifier which is shown in this widget. It is a mandatory field.

4. **Request ID**: This field represents landing screen identifier which is shown in this widget. It is a mandatory field.

5. **Token 4:** This field is not used currently.

6. **Data class**: This CSS class to be applied on the widget User can use CSS auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available CSS class names.

7. **Widget Relative Width**: User can set the width of the Widget relative to the width of entire screen i.e. Width of the widget will be calculated as % of the total width of the screen. User can enter only decimal values and value 1.0 represents the entire screen width.

8. **Widget Relative Height**: User can set the height of the Widget relative to the height entire screen i.e. Height of the Component will be calculated as % of the total height of the screen. User can enter only decimal values and value 1.0 represents the entire screen height.

9. **Widget Relative Position**: User can set the X-Coordinate of the Widget relative to the table in which it is enclosed i.e. X-Coordinate of the Component will be calculated as % of the total width of the table. User can enter only decimal values.

10. **Widget Relative Position Y:** User can set the Y-Coordinate of the Widget relative to the table in which it is enclosed i.e. Y-Coordinate of the Component will be calculated as % of the total height of the table. User can enter only decimal values.

Description of Buttons

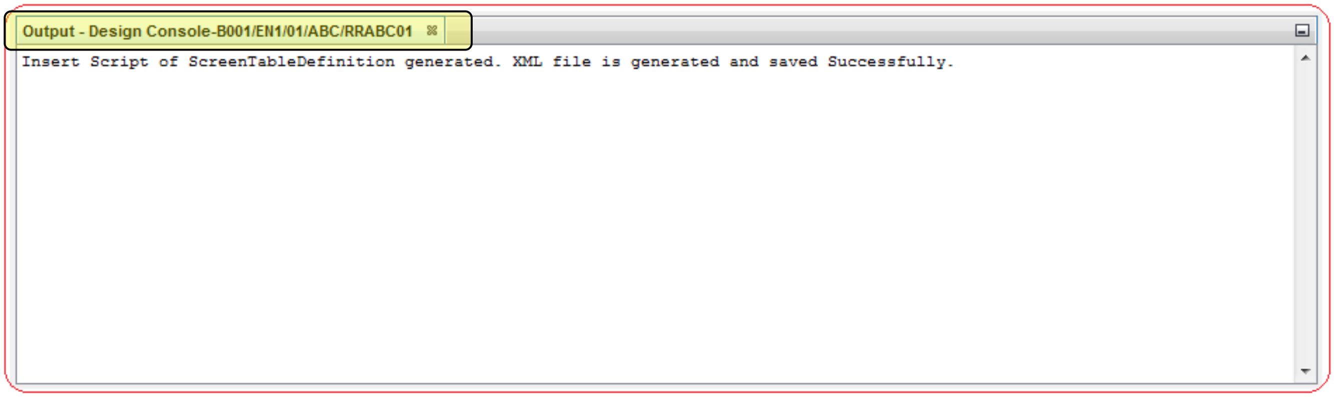
1. 

: Once the user has entered the above details, they need to click on "Apply" button to apply the changes done. If not clicked the changes will not get reflected on revisiting this panel.
2. 

: If the user has not clicked on "Apply" and wants to go back to last applied changes, they can click on "Revert" button.

Console Panel

This panel is at the center-bottom of this tool. This panel, as shown below, is meant to display the messages generated by the tool



As highlighted in the screenshot above the current project details is also displayed, so that a user can identify and distinguish the console messages for different screentemplate projects.

Preview Mobile Application Screen Design

Previewing helps the user to see the screen being designed by them at any stage of the screen design.

Pre-requisites in order to preview mobile screen design for Android Tablets or Phones.

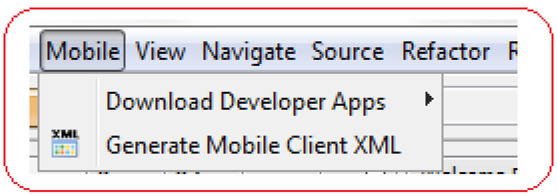
1. **Android Emulator:** User should have Android Emulator installed in their system.

2. **Channel Workbench Preview Server:** User should have the tool's preview server turned ON to preview the screen design. Please Preview Server Menu Item in Menu Bar section for information on how to start preview server.

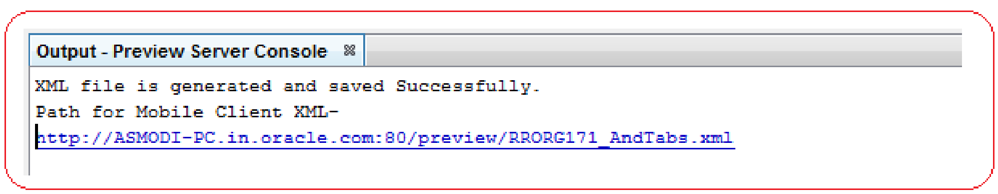
3. **Mobile Client XML:** User needs to generate the mobile client XML for which he wants to preview the screen design.

   Following steps should be followed to generate mobile client XML.

   - ✓ Select the Project from the **ScreenTemplate** Tab in Top Left corner of the tool for which you want to preview the screen design.
   - ✓ Click on **Generate Mobile Client XML** menu item from the Mobile menu in menu bar as shown below,

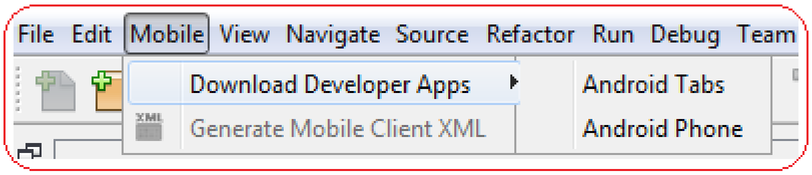
     User can also click on the tool bar shortcut icon provided to generate the mobile client XML.
   - ✓ Client XML is generated and path to xml is displayed to user in output console as shown below,

     Please keep note of this path as this path shall be used to preview the screen design in Android Tablet/Phone application in Emulator.

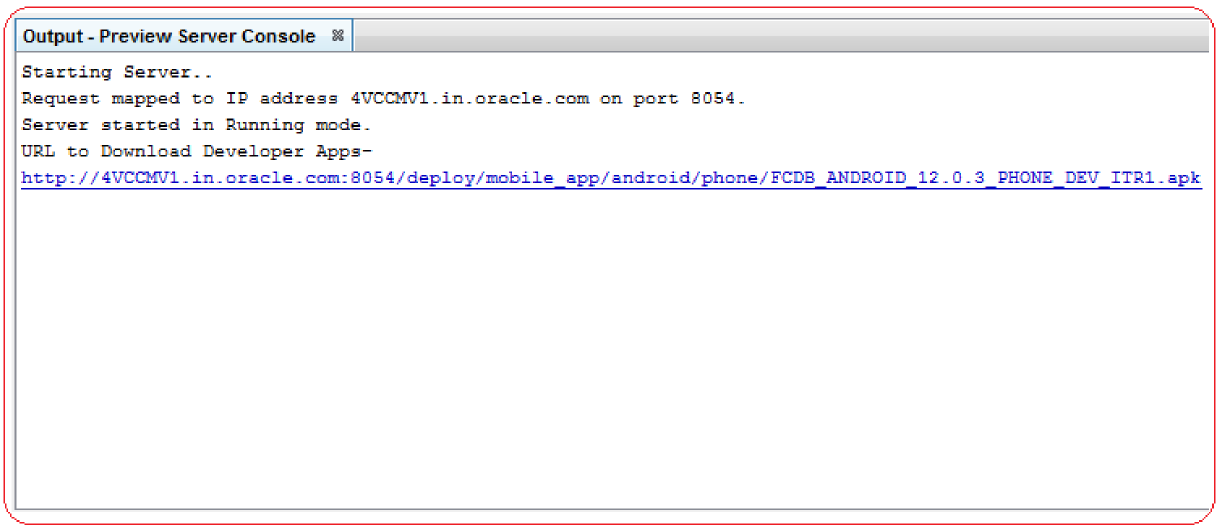
4. **FCDB Android Tablet/Phone Application:** User should have FCDB Android Tablet or Phone Application installed in Android Emulator.

To download the Android Tablet or Phone application, click **on Download Developer Apps** menu item from the Mobile menu in menu bar as shown below,



Select the option form the sub menu (**Android Tabs/Android Phone**) for which you want to download the apk file.

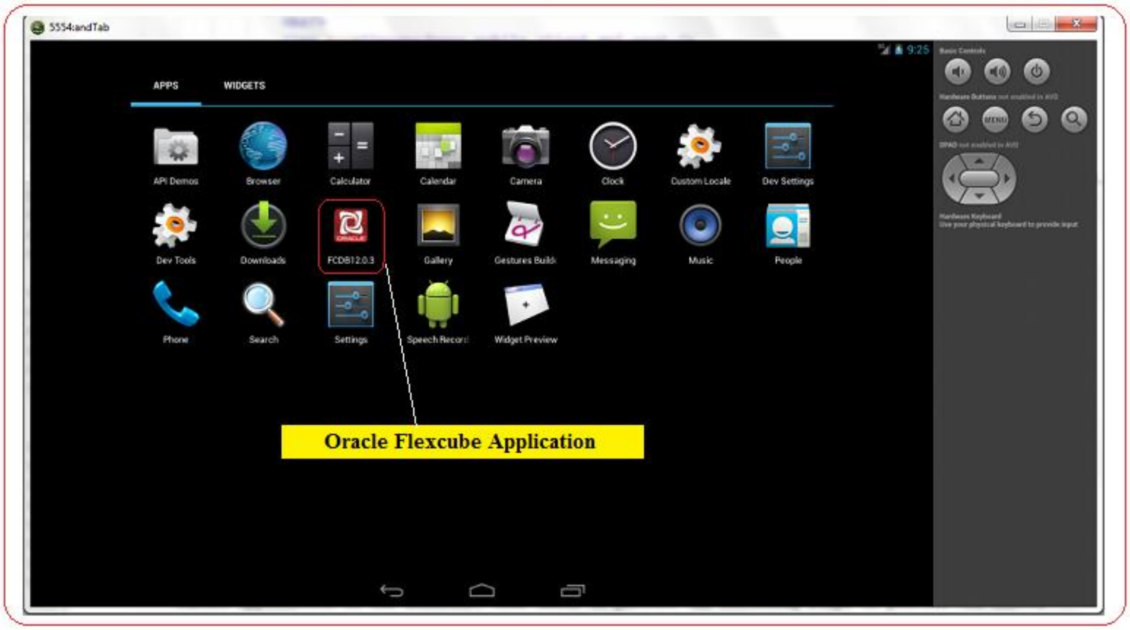
Clicking on the selected option a link will be provided in the server logs to download the apk file as shown below:



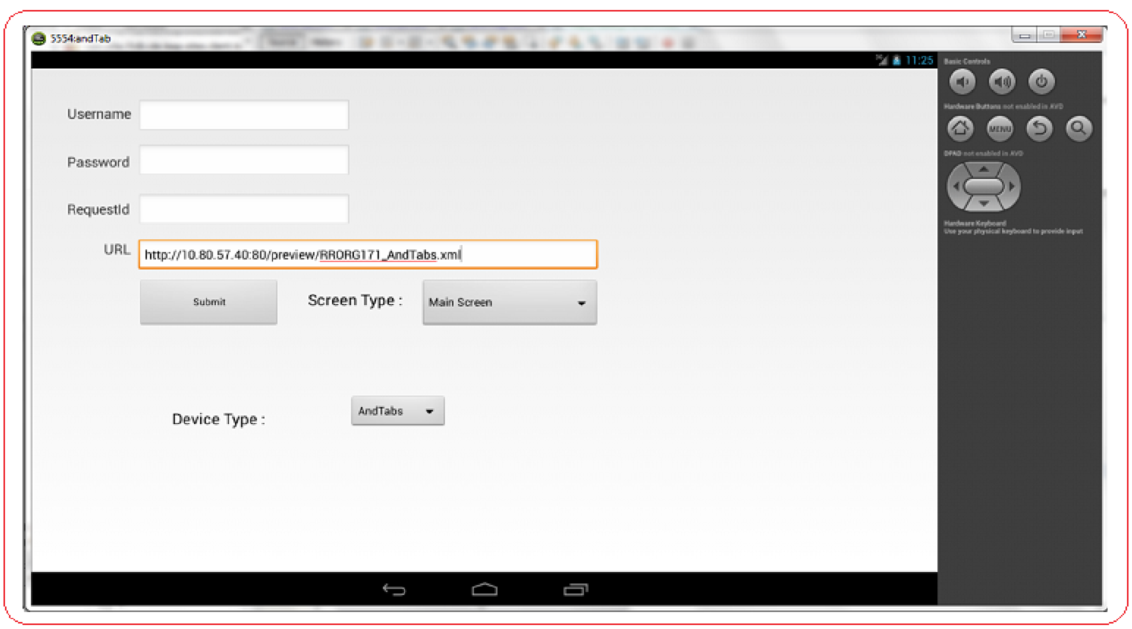
On Android Emulator

After the setup of Android emulator, generating mobile client XML and installing Oracle Flexcube Android Tablet application on the emulator, User needs to perform the following steps to preview the mobile screen design.

1. **Run Oracle Flexcube Application:** Click to run Oracle Flexcube Mobile Banking application in Android emulator.

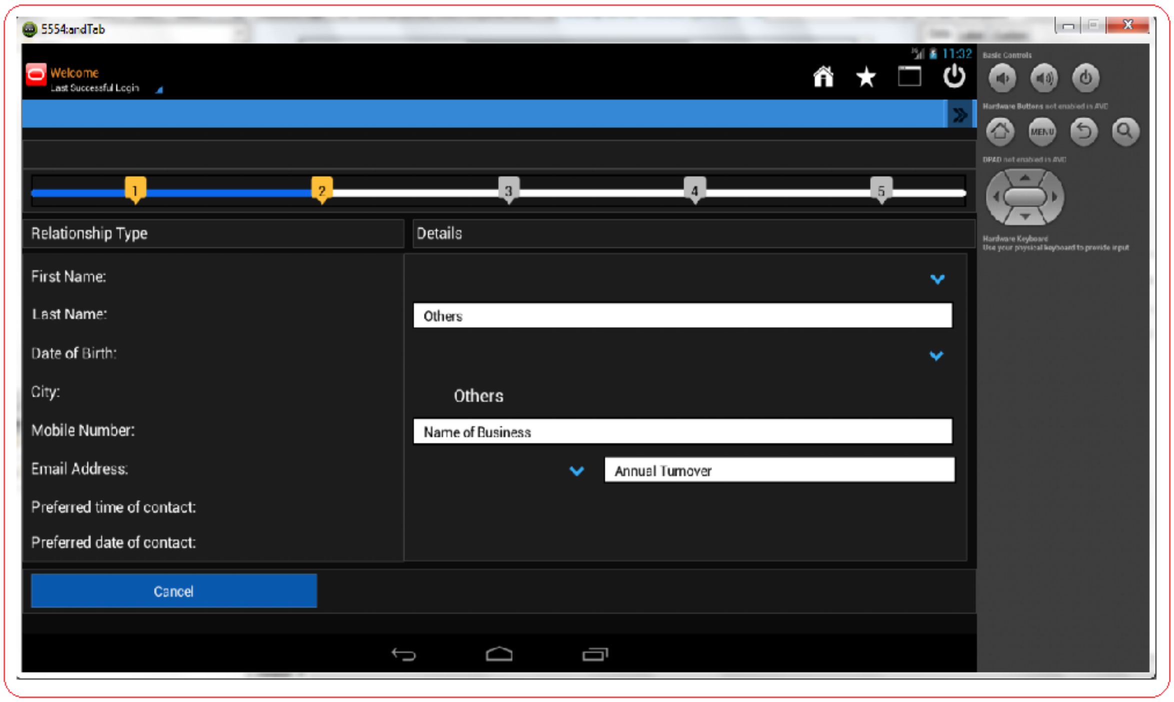


On launching the Oracle Flexcube Banking Application, a splash screen is displayed to user as shown below,



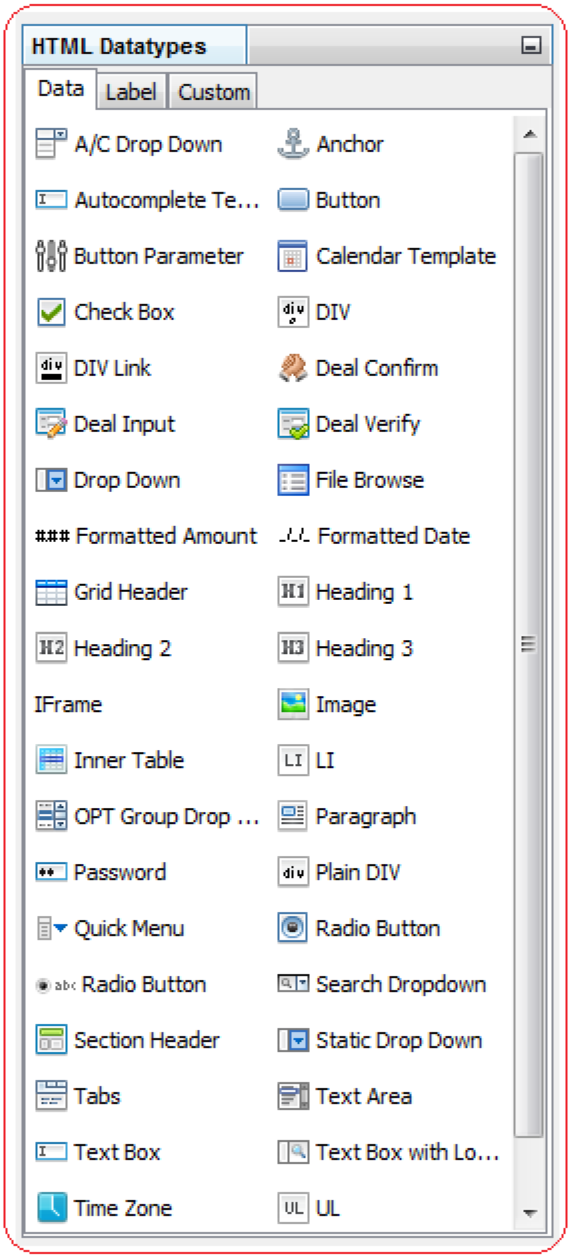
2. **Input Fields:**
   - ✓ **Username**: Clear this input field. The Username should be kept blank.
   - ✓ **Password**: Clear this input field. The Password should be kept blank.
   - ✓ **Request ID**: Clear this input field. The Request ID should be kept blank.
   - ✓ **URL**: In Order to preview mobile screen design, Mobile Client XML hosted by the channel workbench tool is to be given here.
     Enter http://<IP ADDRESS>:<PORT>/preview/xmlname
     For E.g. http://10.180.57.40:80/preview/RRORG171_AndTabs.xml
   - ✓ **Screen Type:** Select Main Screen from Drop Down.
   - ✓ **Device Type:** Select AndTabs/AndPhone from Drop Down.

3. Click on **Submit** to preview your mobile screen design.

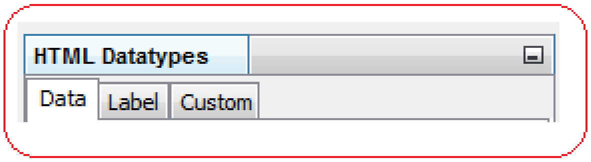


HTMLDatatypes Panel

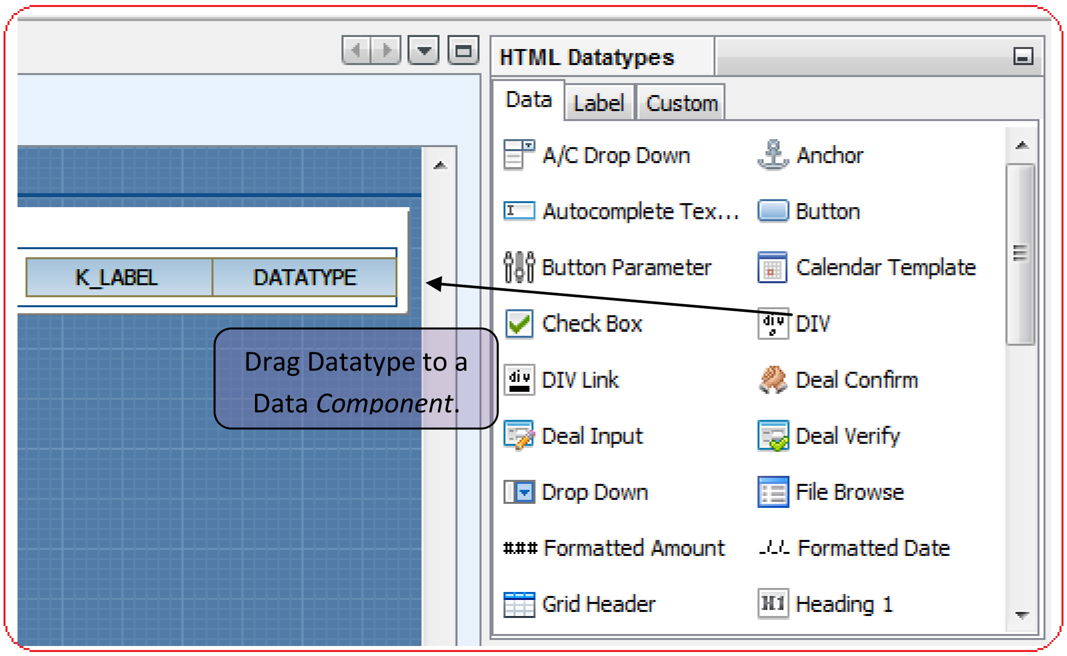
This panel is at the top-right of this tool. This panel contains all the HTMLDatatypes currently supported in Development Workbench framework in the form of button's, as shown below. User can click on these buttons to use these HTMLDatatypes to design the screen.



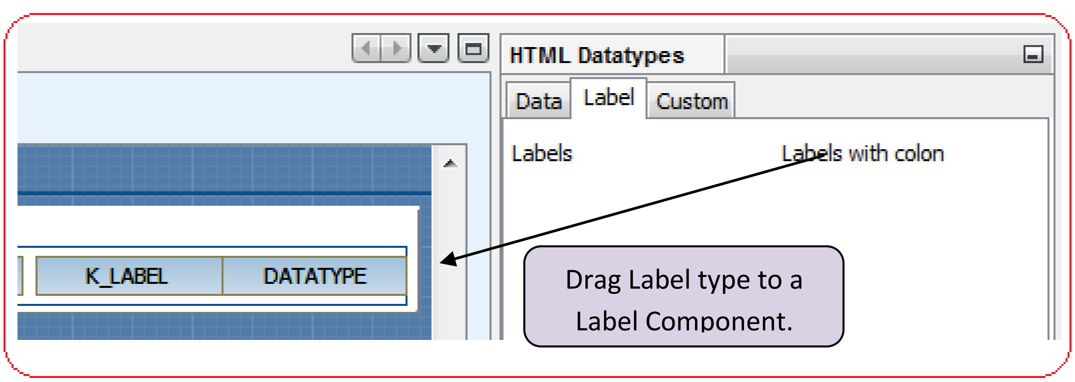
These HTMLDatatypes has been classified into three groups.



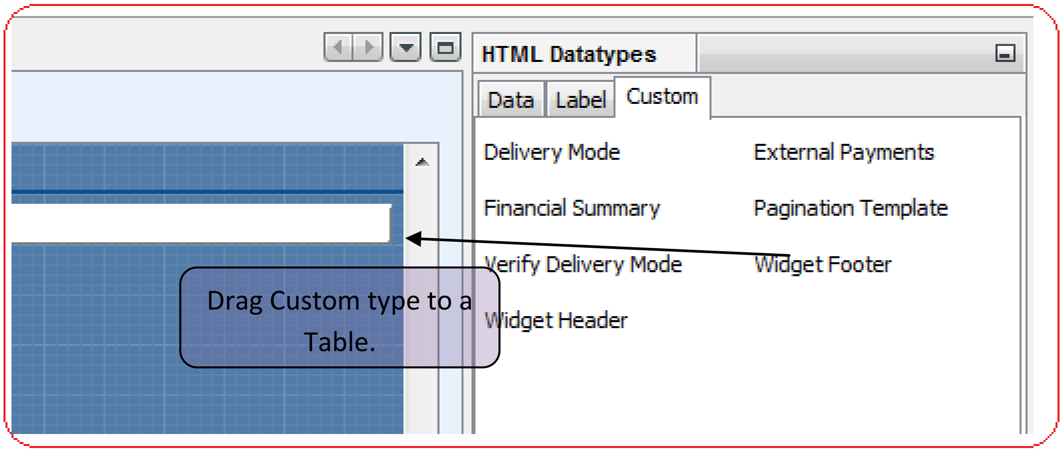
1. **Data:** These datatypes are those datatypes which helps user to place certain HTML components like textbox, dropdown inside a particular column of a particular row of particular HTML Table.



2. **Label:** These datatypes help user to configure the labels in the form. Currently there are two configurations available. One is “Labels” which can be used for simple label text with mandatory icon, if any and other is “Labels with colon” which can be used for simple label text with mandatory icon, if any and a colon appended at the end.



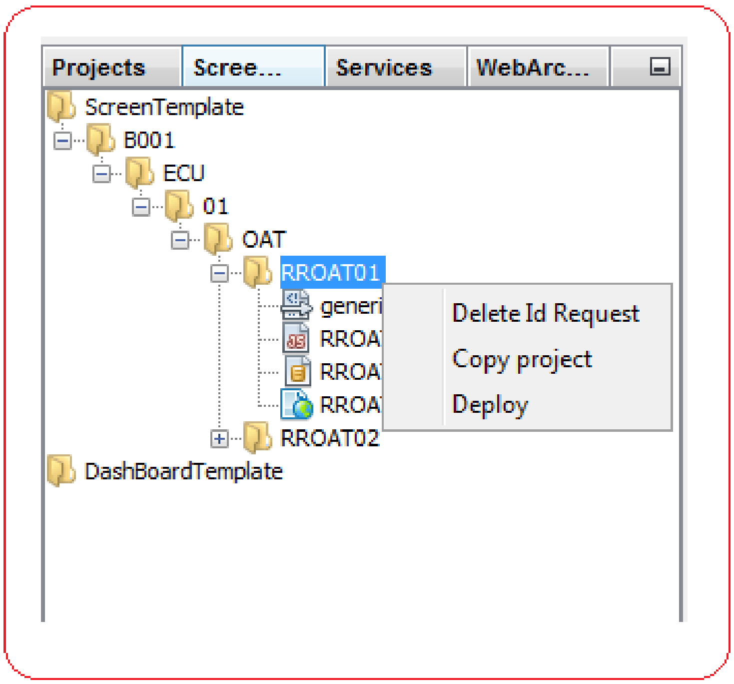
3. **Custom:** These datatypes are nothing but a predefined layout which one can directly place at certain position of the screen. By predefined layout it means that rather than defining each component of the screen one can call these datatype templates to place an html layout at a particular table sequence. As these layouts can be used at a particular table sequence which means these datatypes can be used while defining the HTML Table attributes. Clicking these datatypes sets the value in "Customized Template ID" drop down of "HTML Table Details" work area panel.



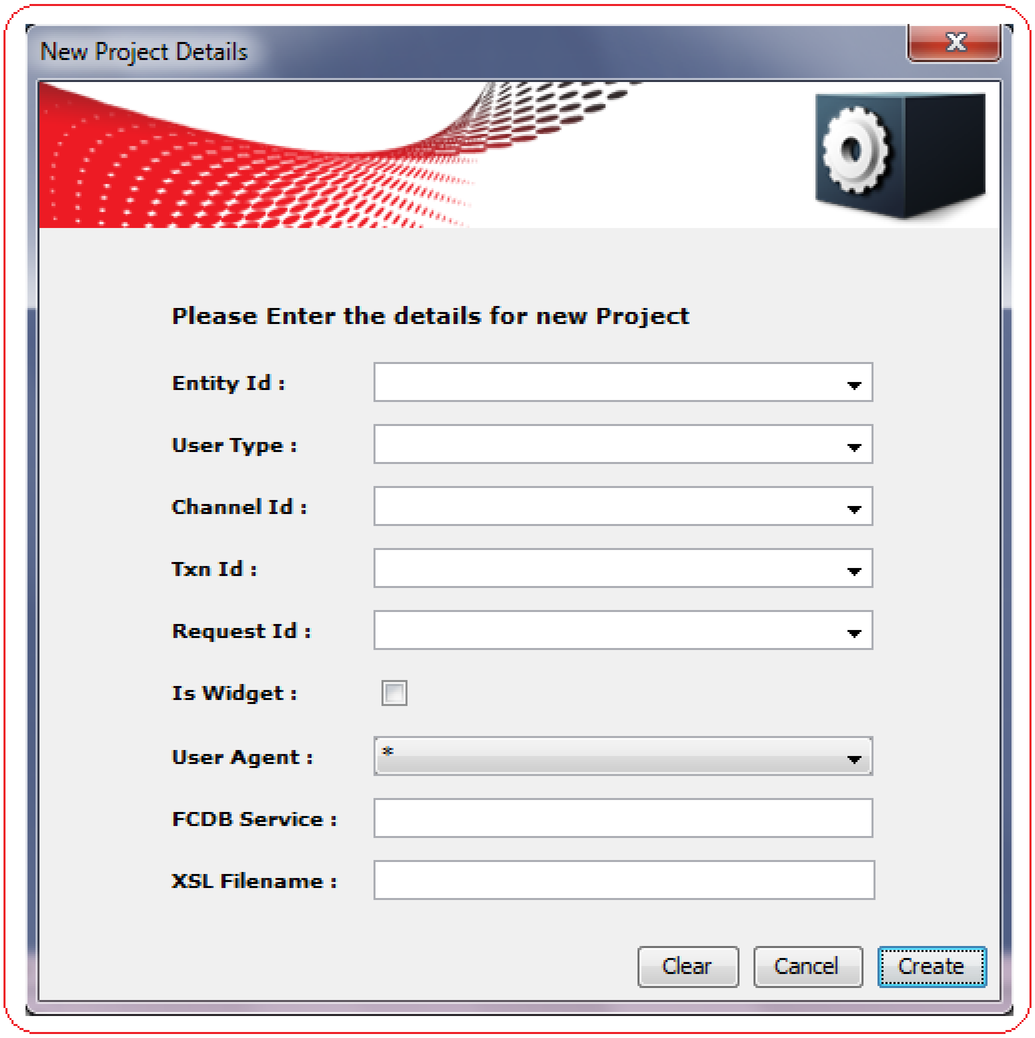
Deleting / Copying / Deploying Project

1. Copy Project

To Copy an existing project from the workspace, click on IdRequest node (at the top left of the tool). A menu appears with the options **'Delete Id Request'**, **'Copy Project'** and **'Deploy'** as shown below.



To Copy project click on **'Copy Project'** option. Copy Project Dialog will be opened as shown below.



After entering the appropriate values, click on create button. Create button will create a new project in the workspace, and all the content of selected project will be copied to the newly created Project.

The project details will get loaded in the workspace layout tree (at the top-left of this tool). Four files namely, IdRequest.js, IdRequest.sql, IdRequest.xml and XXXX.xsl will be created in the workspace "ScreenTemplate".

2. Delete Project

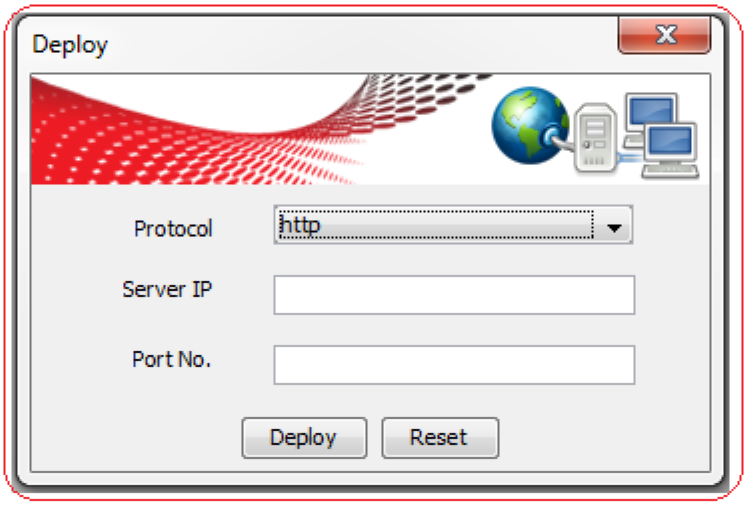
To delete an existing project from workspace, click on IdRequest node (at the top left of the tool). A menu appears with the options **'Delete IdRequest'**, **'Copy Project'** and **'Deploy'**.

To delete project click on **'Delete IdRequest'** option. 'Delete IdRequest' option will delete the selected IdRequest along with all the parent directories of the selected idRequest from the workspace "ScreenTemplate".

3. Deploy Project

To deploy an existing project in workspace, click on IdRequest node (at the top left of the tool). A menu appears with the options **'Delete IdRequest'**, **'Copy Project'** and **'Deploy'**.

Clicking the **"Deploy"** option will open a dialog as shown below.



Enter the Protocol (http/https), Server IP and Port No. and click on "Deploy" to deploy the project on the given server.

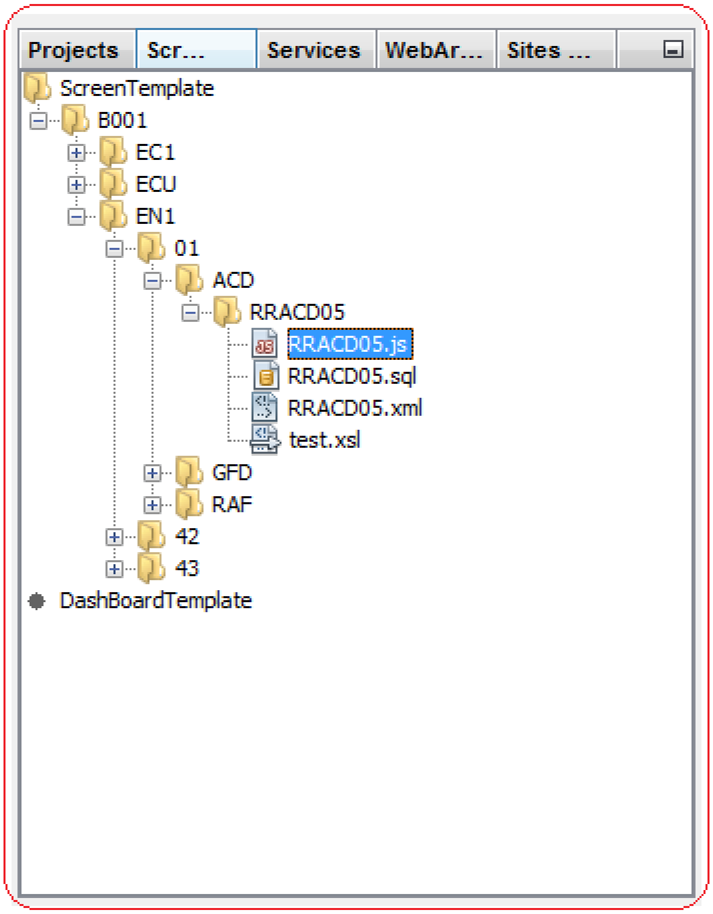
JavaScript / CSS / Image Editor

1. JavaScript Editor

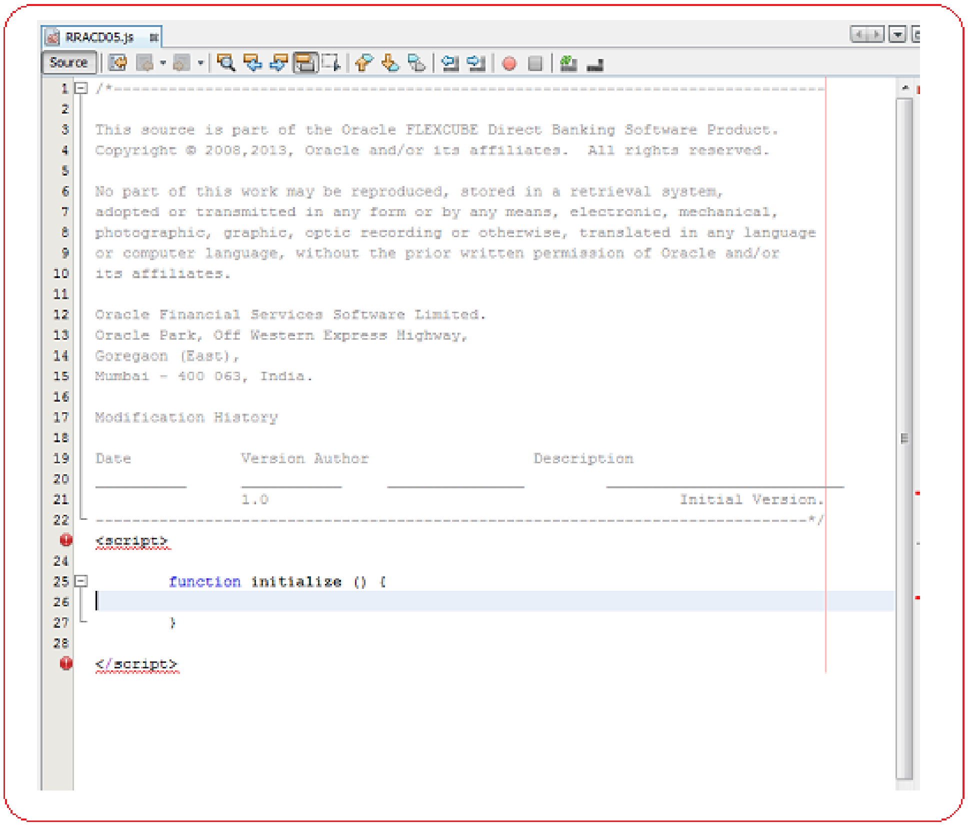
For each screen designed a developer has to write corresponding JavaScript file. The name of the js file must be in the format <idrequest>.js like (RRTFT01.js).This js file must have "initialize()" method by default, even though this method is not required, a developer has to write a blank implementation of this method in the js file.

The functions which are being assigned to the Datatypes while doing the Screen Design needs to be implemented in this file. The .js file can contain standard FCDB language key words (like "K_").

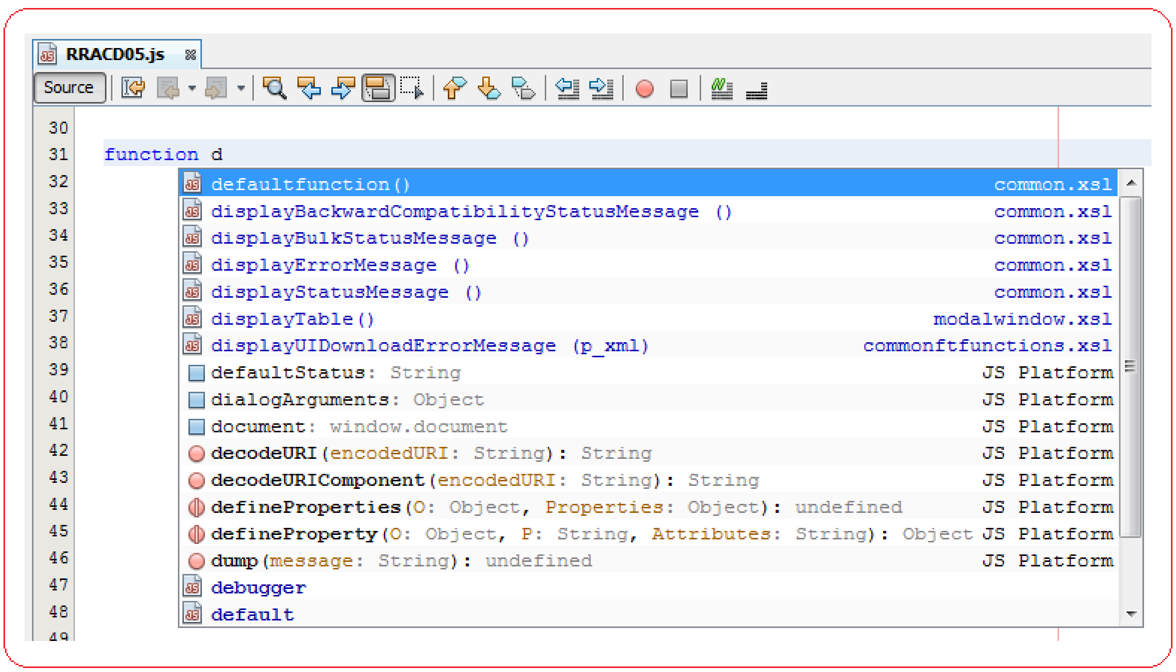
This tool will give you the option to write your js file. For each Screen Template project in the top-left panel a <idrequest>.js tree node is created as shown below.



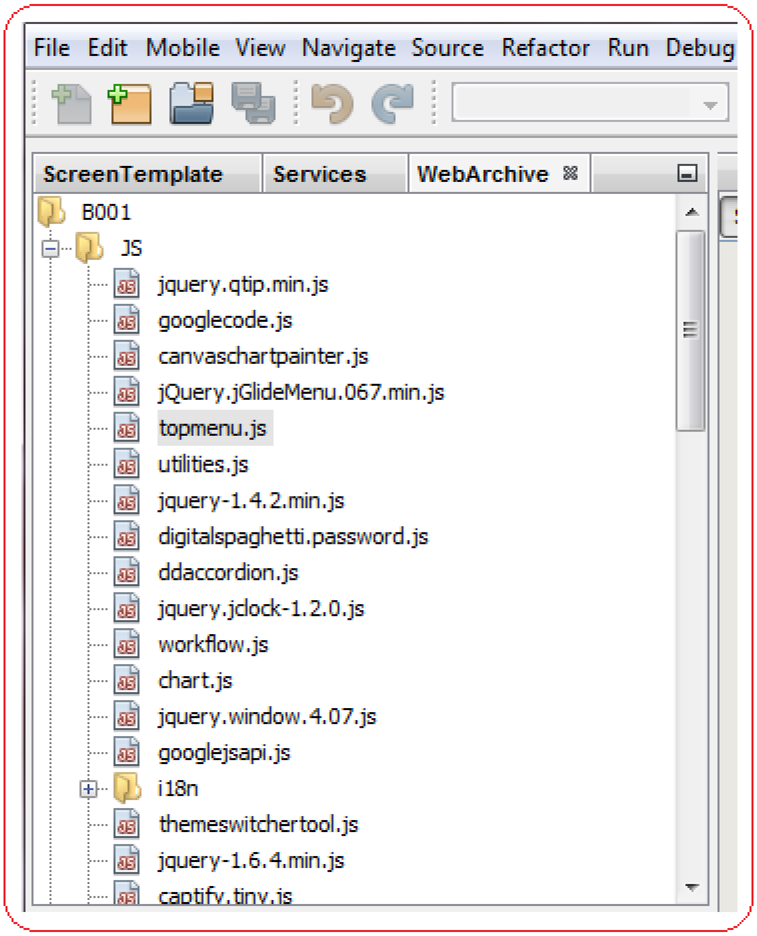
Double clicking the tree node will open the ".js" file in JavaScript editor as shown below.



The JavaScript editor provides auto-suggest feature using **CTRL+SPACE**. The list of auto-suggest includes JS functions from the tool as well as JS Platform.

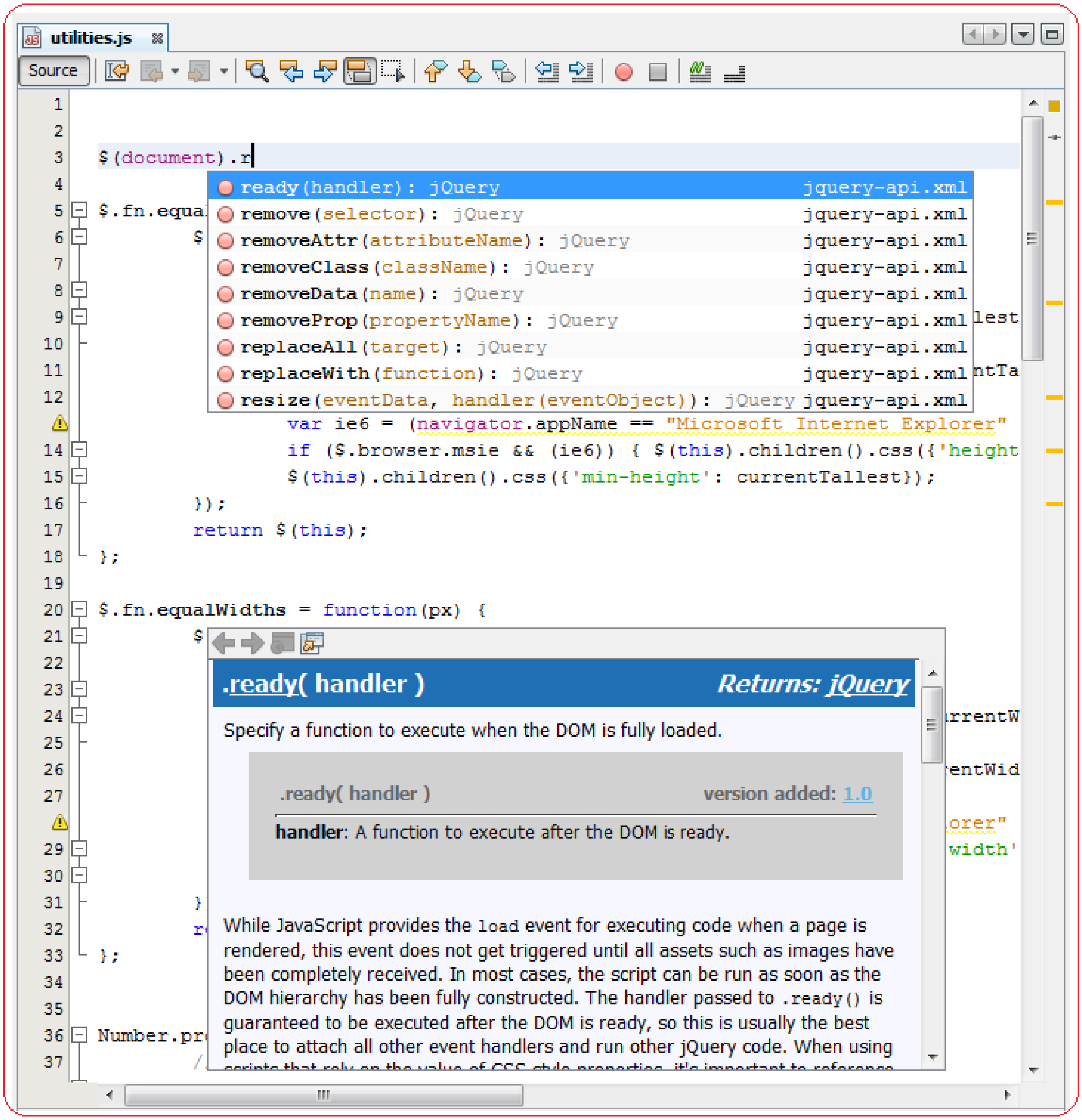


User can also edit JS files available in "Web Archive" WAR by double clicking on a JS file in the Web Archive window as shown below.



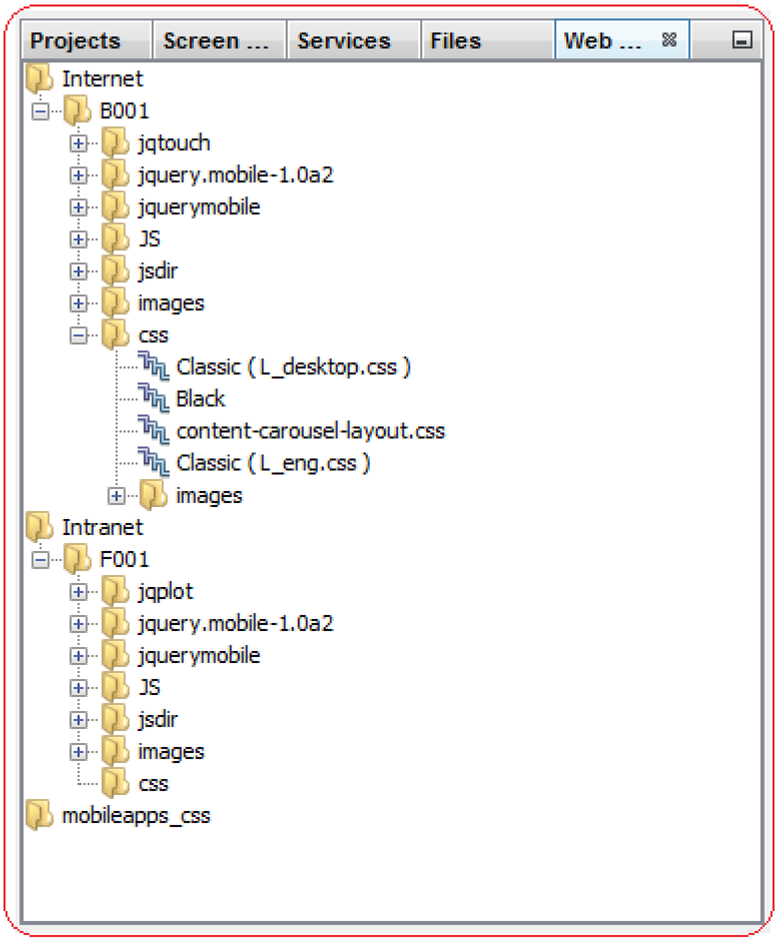
The JavaScript editor of the tool also provides **JQuery** functions as auto-suggest. The user can select from the list of available JQuery functions.

Documentation also appears for the JQuery function highlighted by the user as shown below.


utilities.js
Source
ready(handler): jQuery jquery-api.xml
remove(selector): jQuery jquery-api.xml
removeAttr(attributeName): jQuery jquery-api.xml
removeClass(className): jQuery jquery-api.xml
removeData(name): jQuery jquery-api.xml
removeProp(propertyName): jQuery jquery-api.xml
replaceAll(target): jQuery jquery-api.xml
replaceWith(function): jQuery jquery-api.xml
resize(eventData, handler(eventObject)): jQuery jquery-api.xml
.ready(handler) Returns: jQuery
Specify a function to execute when the DOM is fully loaded.
.ready(handler) version added: 1.0
handler: A function to execute after the DOM is ready.
While JavaScript provides the load event for executing code when a page is rendered, this event does not get triggered until all assets such as images have been completely received. In most cases, the script can be run as soon as the DOM hierarchy has been fully constructed. The handler passed to .ready() is guaranteed to be executed after the DOM is ready, so this is usually the best place to attach all other event handlers and run other jQuery code. When using

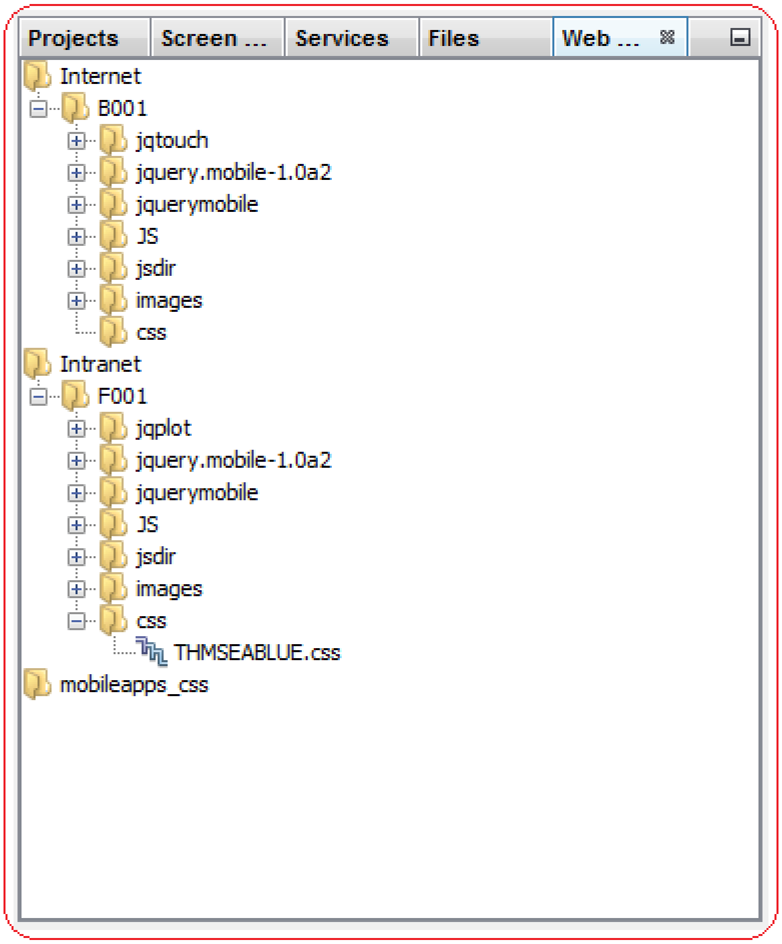
2. CSS Editor

The CSS themes available for a particular User Type & Channel Id (Internet 01/Mobile Browser 42) are displayed under the **"css"** folder of "Web Archive" window. The CSS Tree is shown below.



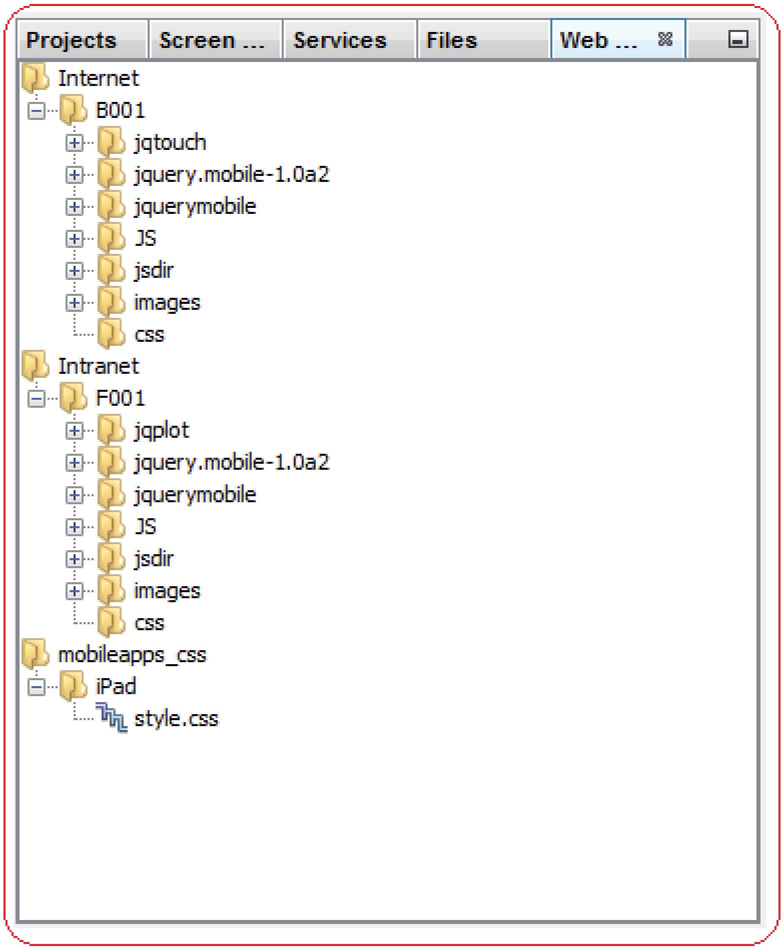
The CSS list for **Intranet Channel (11)** is displayed under the "**Intranet"** folder of "Web Archive" window. "Intranet" folder displays CSS list only when Screen Design project with Channel Id Intranet channel 11 is selected.

The CSS Tree for mobile applications (11) channel is shown below.



The CSS list for **Mobile Applications Channel (43)** is displayed under the "**mobileapps_css"** folder of "Web Archive" window. "mobileapps_css" folder displays CSS list only when Screen Design project with Channel Id Mobile Applications channel 43 is selected.

The CSS Tree for mobile applications (43) channel is shown below.



CSS Theme Switching

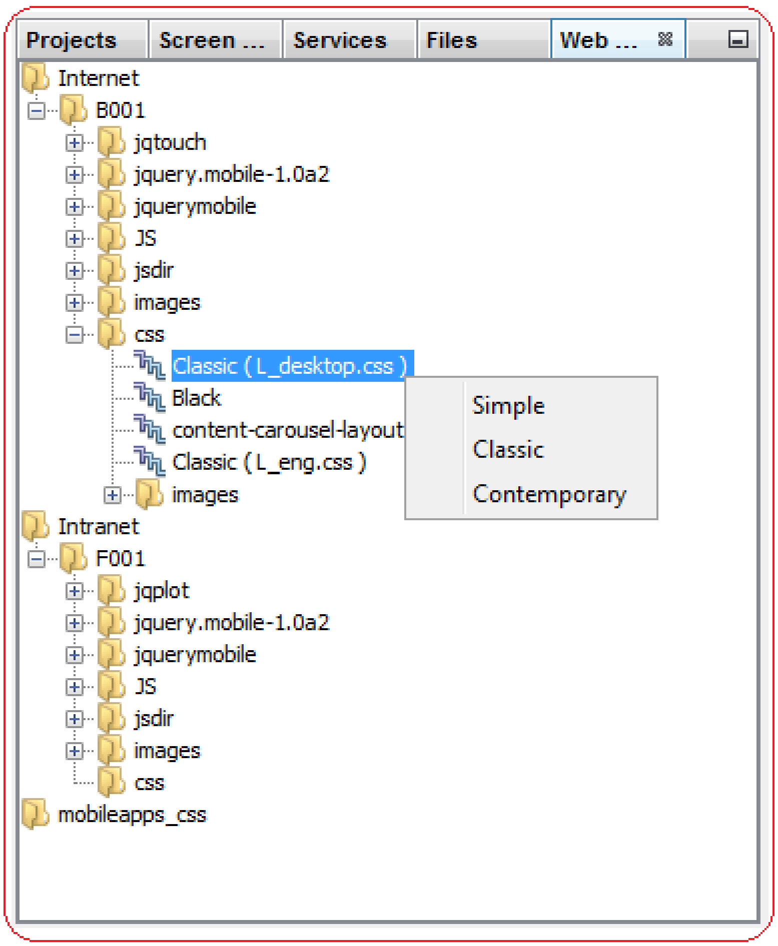
1. Internet Channel (01)

CSS themes for Internet Channel are available in two ".css" files.

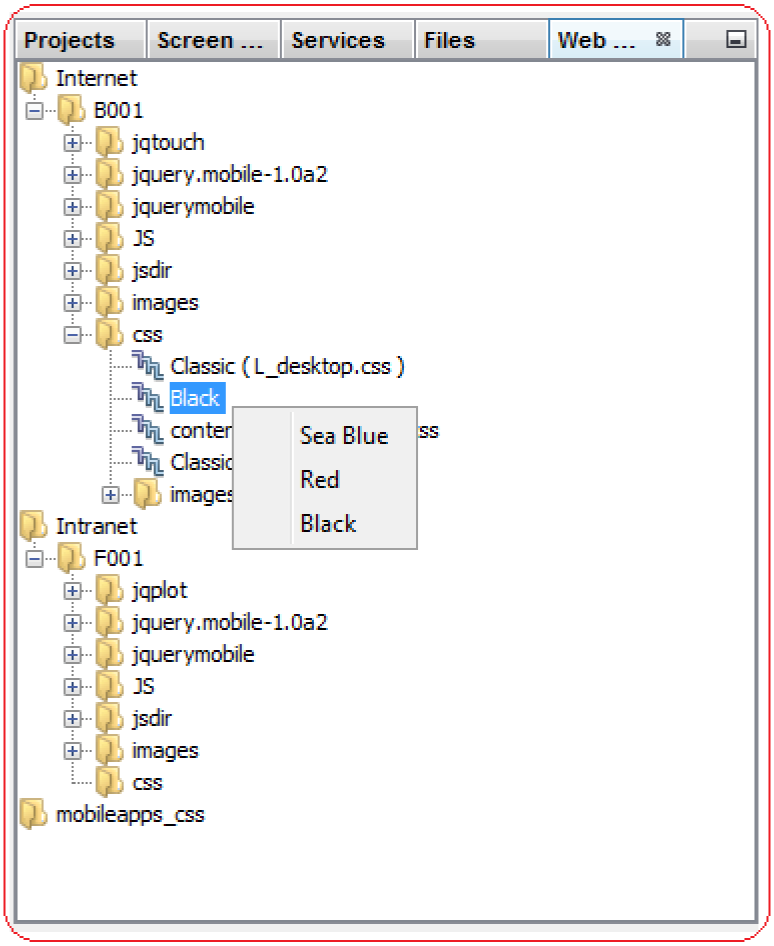
i. Layout theme { Simple | Classic | Contemporary | Metro }

ii. Color theme { Color Options based on Layout theme }

To change Layout theme, right click on current Layout theme and select one from the available options as shown below.

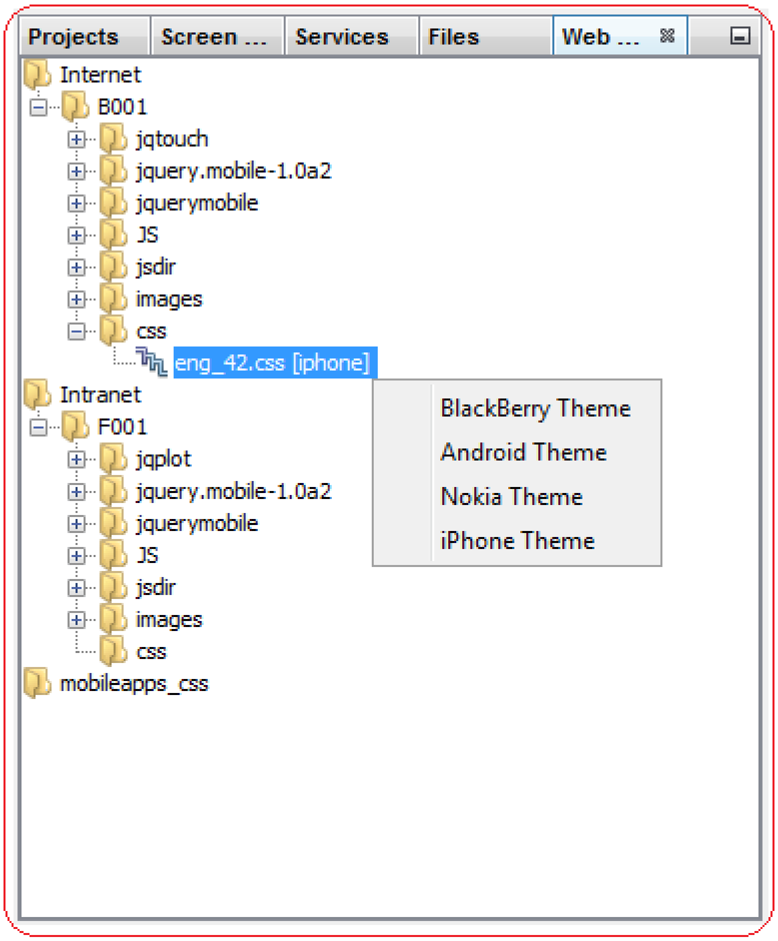


To change Color theme, right click on current Color theme and select one from the available options as shown below.

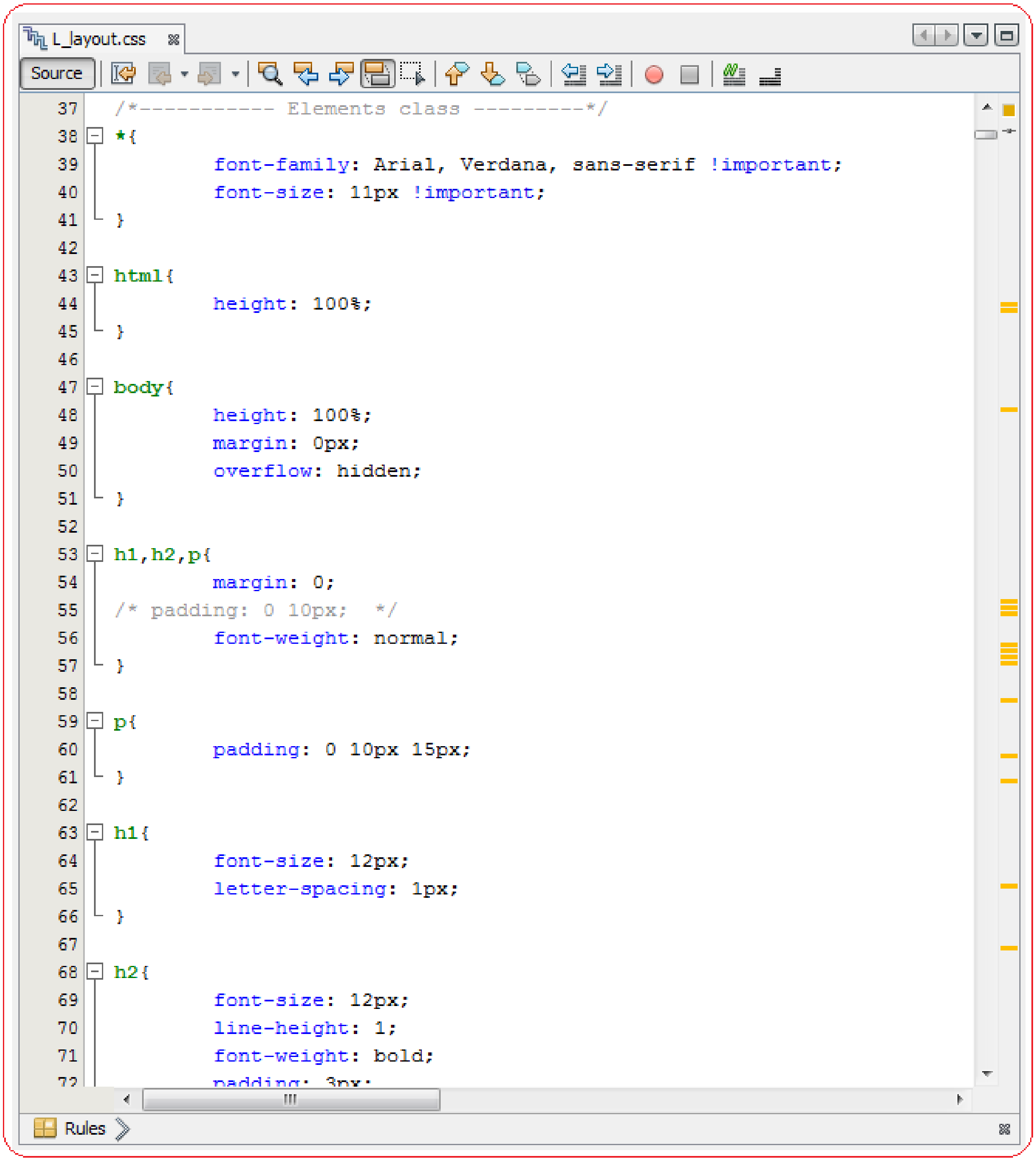


2. **Mobile Browser Channel (42)**

To change Mobile browser channel theme, right click on current theme and select one from the available options as shown below.

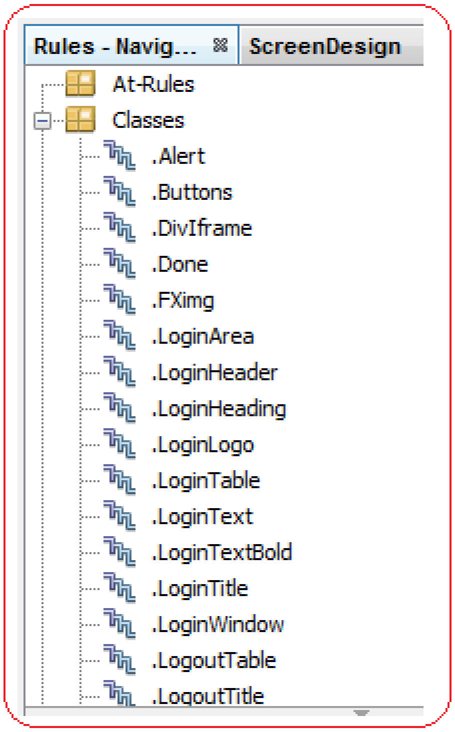


Double click on the CSS theme to open it in CSS editor.



The CSS editor is full-featured editor. To know more of its features, press “**F1**” on opened CSS editor file.

All the classes in a CSS file are loaded in **“Navigator”** window in the bottom left panel of this tool. It will allow user to navigate through the available CSS classes.



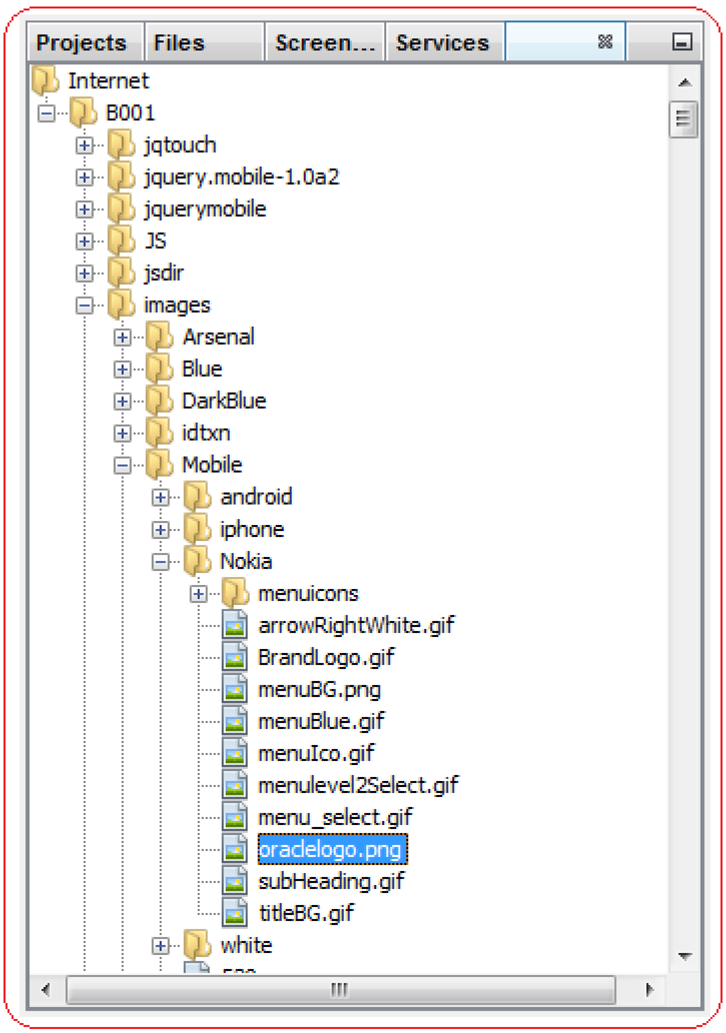
Click on any CSS class to view or edit it.

3. Image Editor

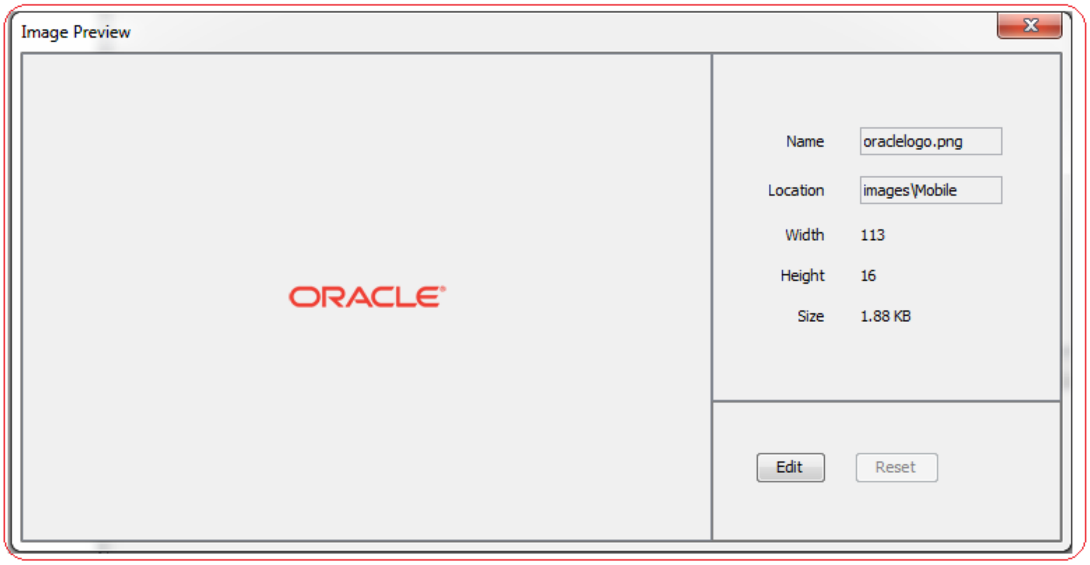
All the images which are available in **"images"** folder of Web Archive WAR (for e.g. B001.WAR) will be displayed as a tree structure under the images folder of "Web Archive" window.

Moreover the images available for a particular theme of Internet Channel (Channel Id 01) are available in the "**css**" folder.

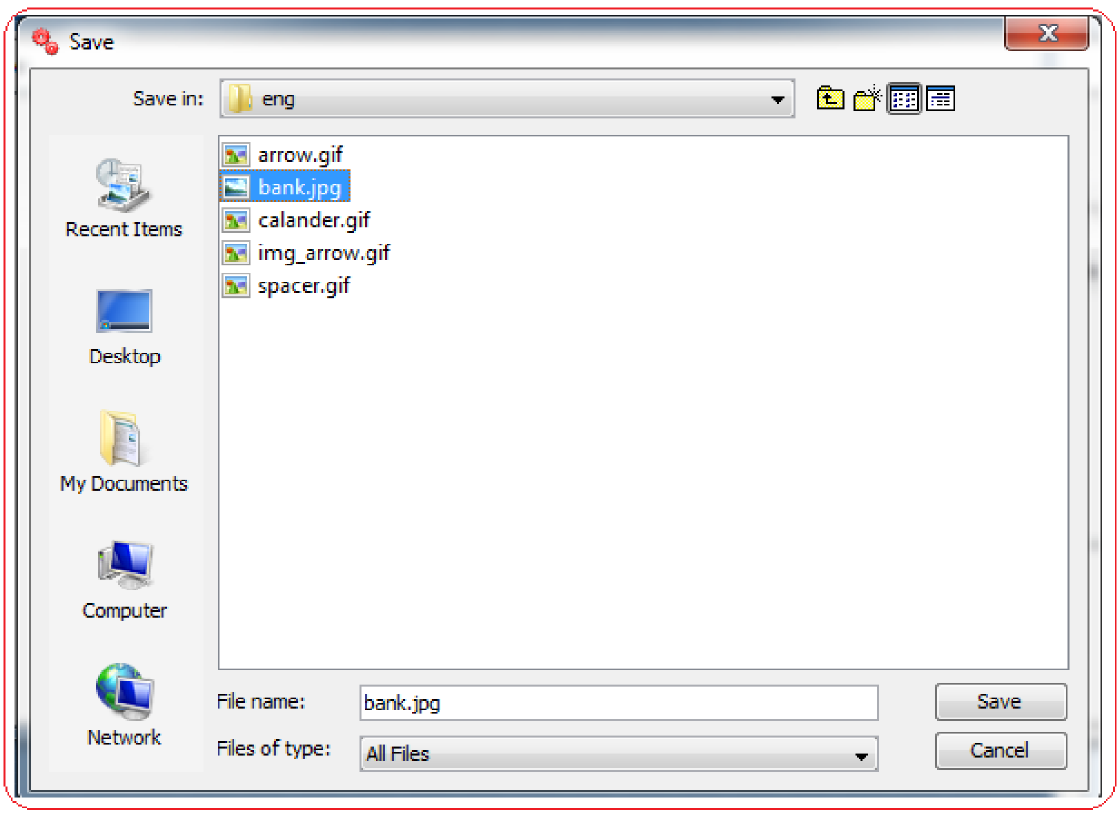
The Image Tree is shown below:



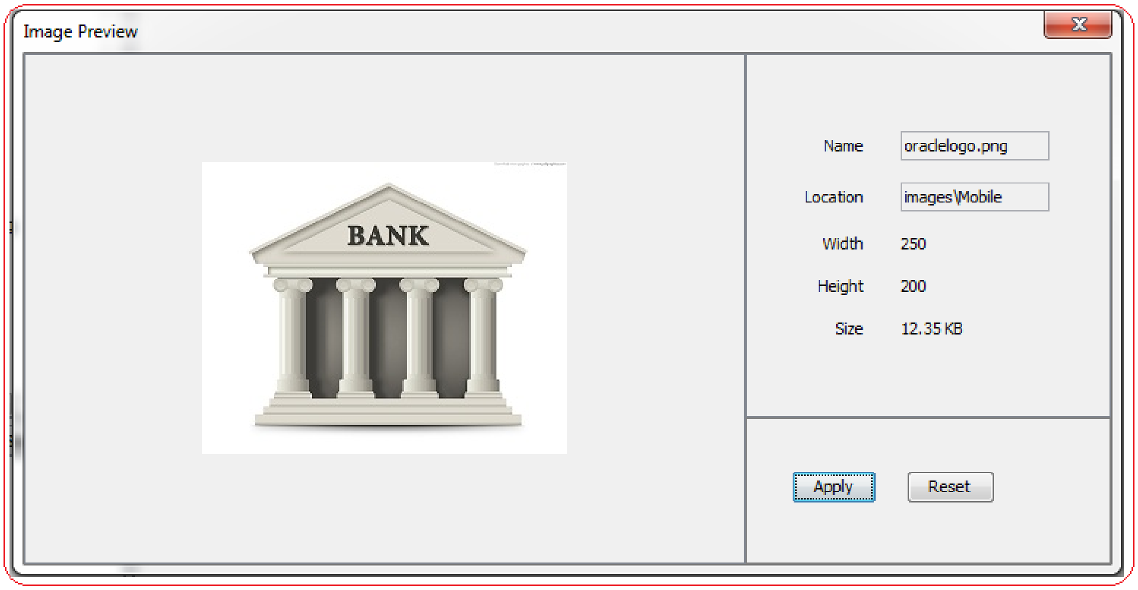
Double click on the image to preview.



User can edit the image. Click on **"Edit"** button to edit the image, a file chooser dialog will be opened as shown below.

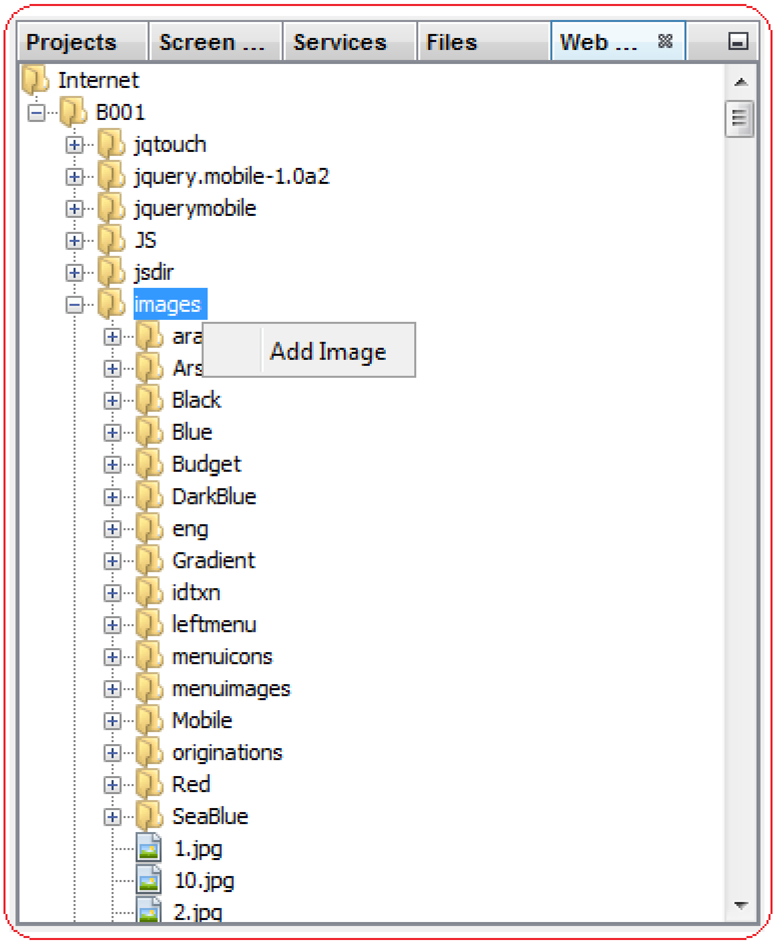


Select an image and then say "Apply". The image will be replaced by the selected image.

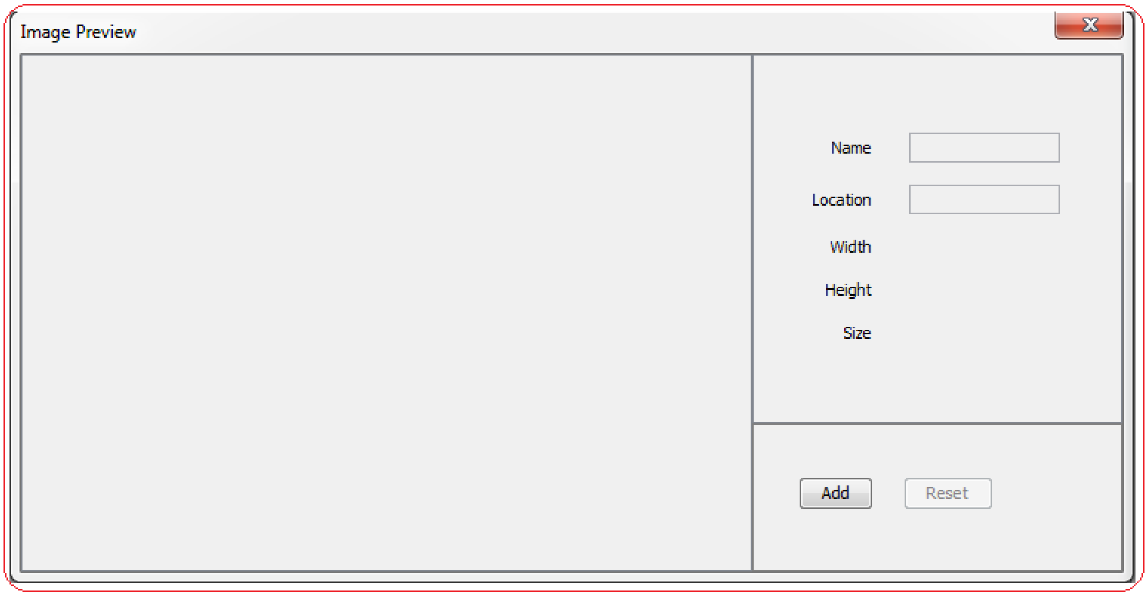


Adding an Image

User can add an image by right clicking on the **"images"** folder and selecting **"Add Image"** option as shown below:



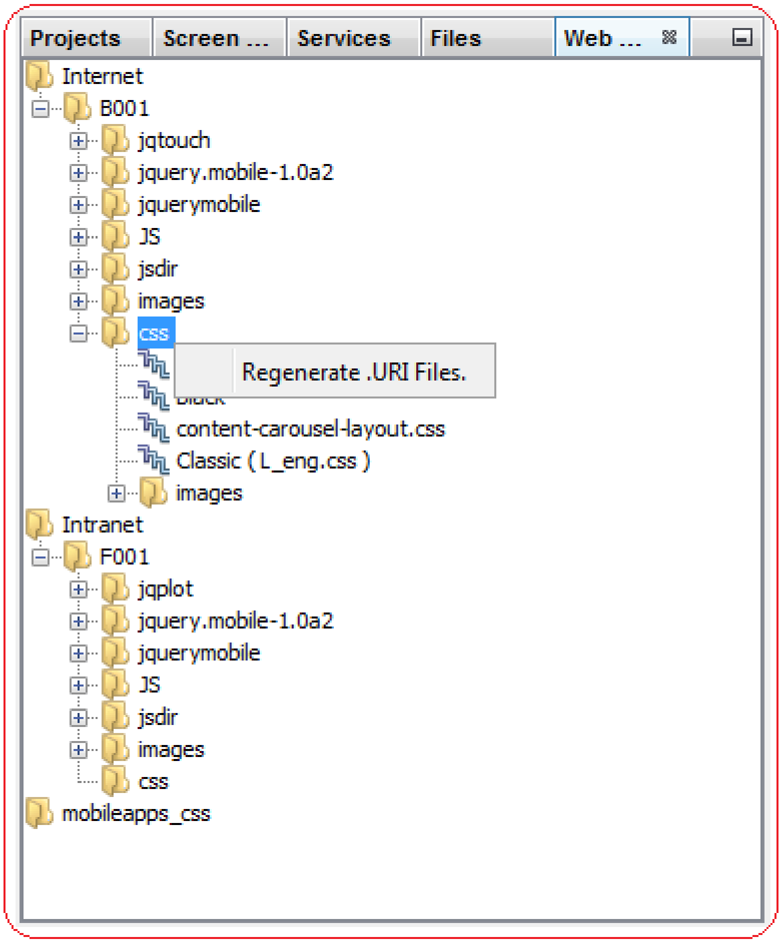
Click on "Add" button as shown below:



A file chooser dialog will be opened to browse the new image path. Enter proper image name and path. Click on "Apply" button once all details are filled.

If an image is edited / replaced, it is necessary to regenerate the Data URI CSS files (".uri.css"). The Data URI CSS files stores Base64 representation of the image in the ".uri.css" files in which the image is used.
User can preview screen designed with updated image post Data URI CSS file conversion.

To perform this operation, right click on the menu option of "css" folder as shown below:

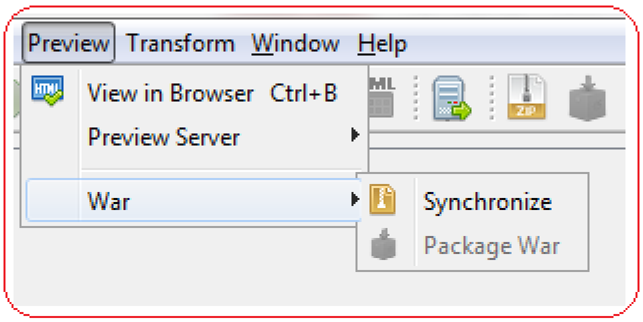


Synchronize / Package WAR

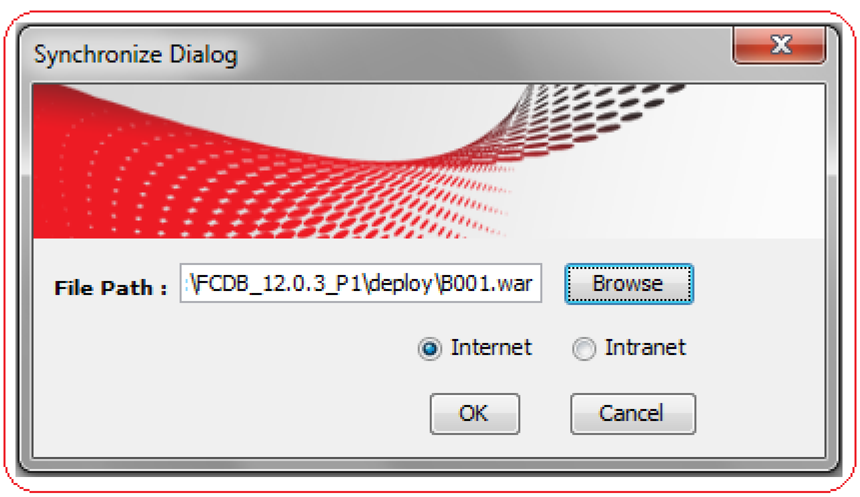
1. Synchronize WAR

To ensure that one have the latest CSS classes and Images loaded in Development Workbench tool before editing such web components, one can synchronize the WAR bundled in tool using this option.

Select War option from the Preview Menu. A sub-menu **"Synchronize"** will be available, as shown below:



Clicking on the "Synchronize" option, a dialog box will open where user can enter the path of the war file to be synchronized as shown below:



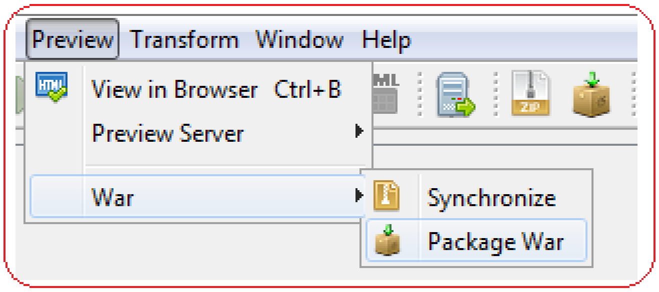
Select the radio button corresponding to the channel (Internet (01)/Intranet (11)) for which you want to synchronize the war file.

Clicking the ok button will synchronize the WAR bundled in the tool.

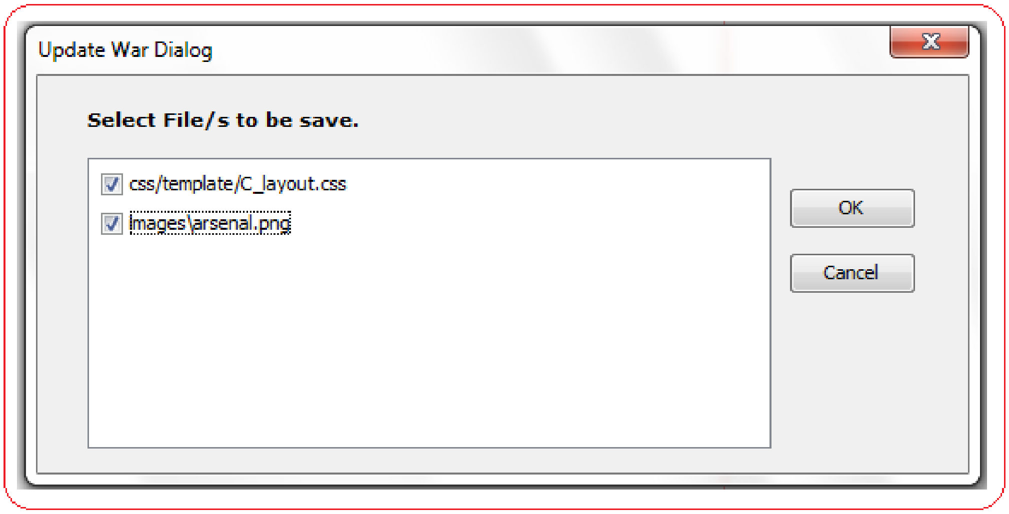
2. Package WAR

Package war will help user to repackage the war with updated CSS and image files. Package War will be enabled only if any Web component like CSS or images has been edited by the user.

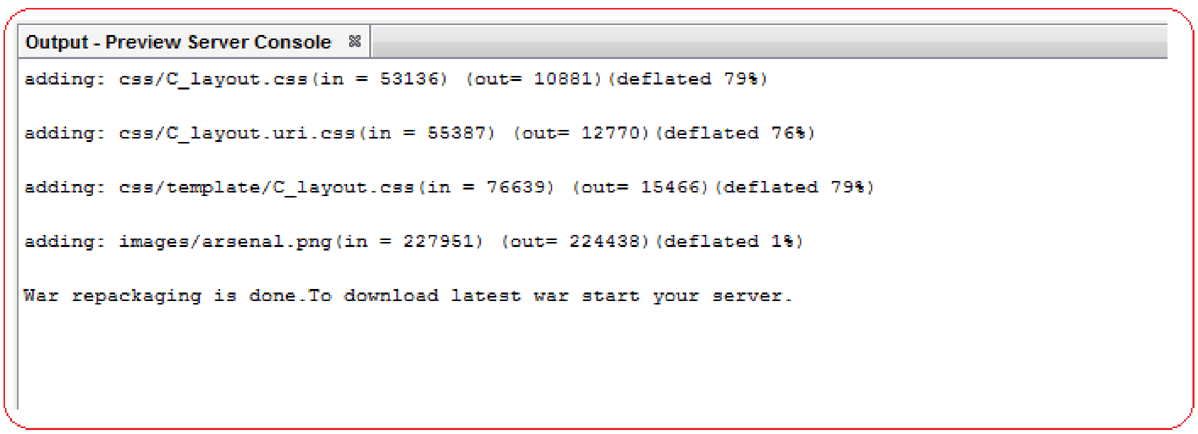
Select War option from the Preview Menu. A sub-menu **"Package War"** will be available, as shown below:



Clicking on the package war, a dialog with the updated files will open which will let user choose the changes to be made in the war as shown below:

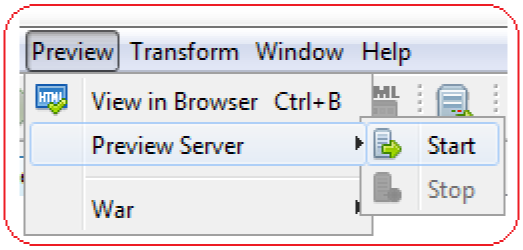


Clicking the OK button will package the war with the selected changes and logs will be generated for the same in server logs as shown below:

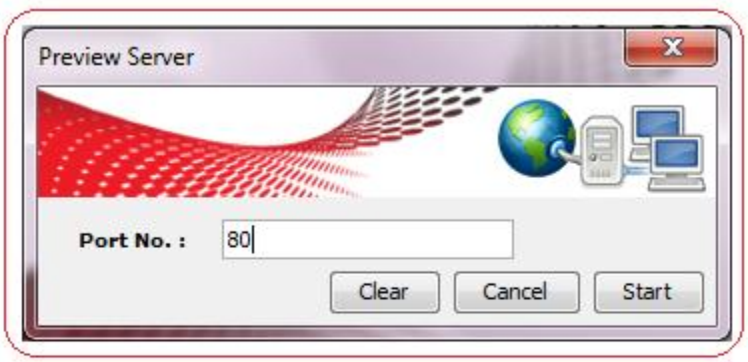


Download Latest WAR

To download the latest packaged war, start the server by selecting **Preview Server** option from Preview Menu. A "Start" sub-menu will be available as shown below:



Clicking on start a dialog will open. Enter a valid port no. (i.e., it should be a 4 digit numeric value) as shown below:



Clicking on the start button the server will start and a link will be provided in the server logs to download the packaged war as shown below:

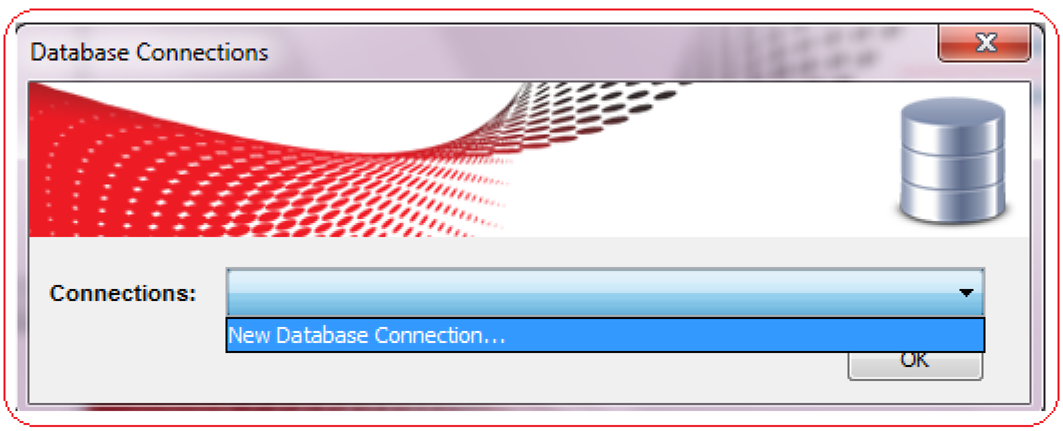


Database Connection

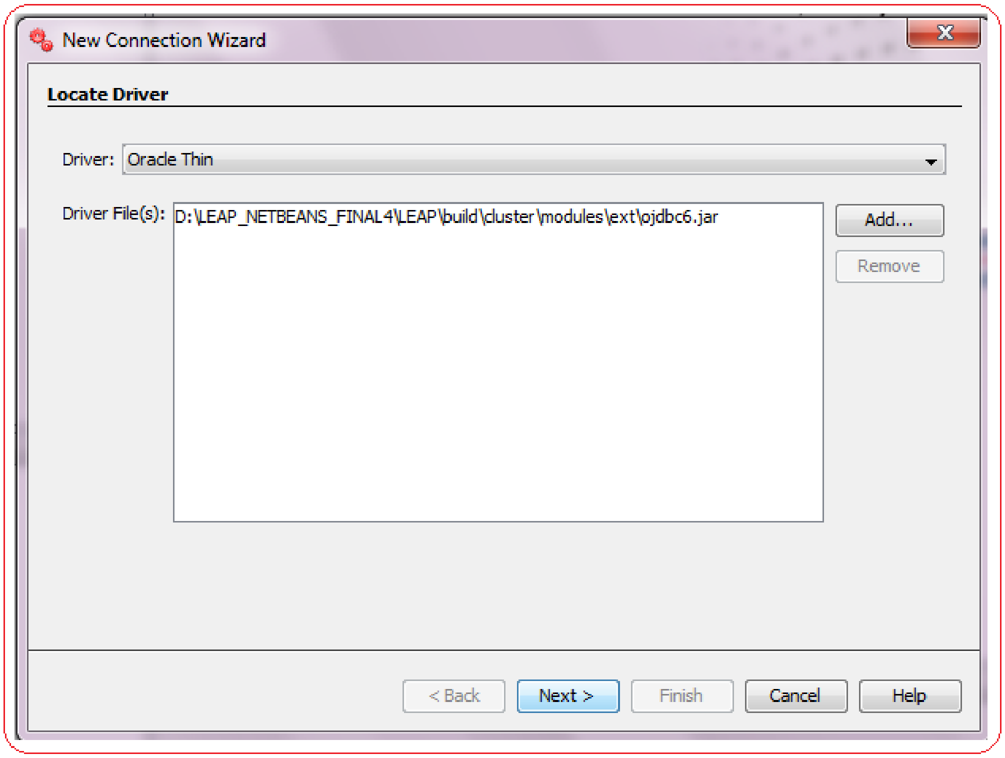
Database Connection provides a feature to connect the tool to a database schema, and allows us to configure different FCDB configuration parameters. Configuration Wizards have been provided using which user can search for an existing configuration in the database, and can add new configuration in database in a step-wise manner. These wizards will generate the Add/Update/Delete scripts depending upon the action taken by the user and also allow user to execute them against the selected database.

Connecting to the Database

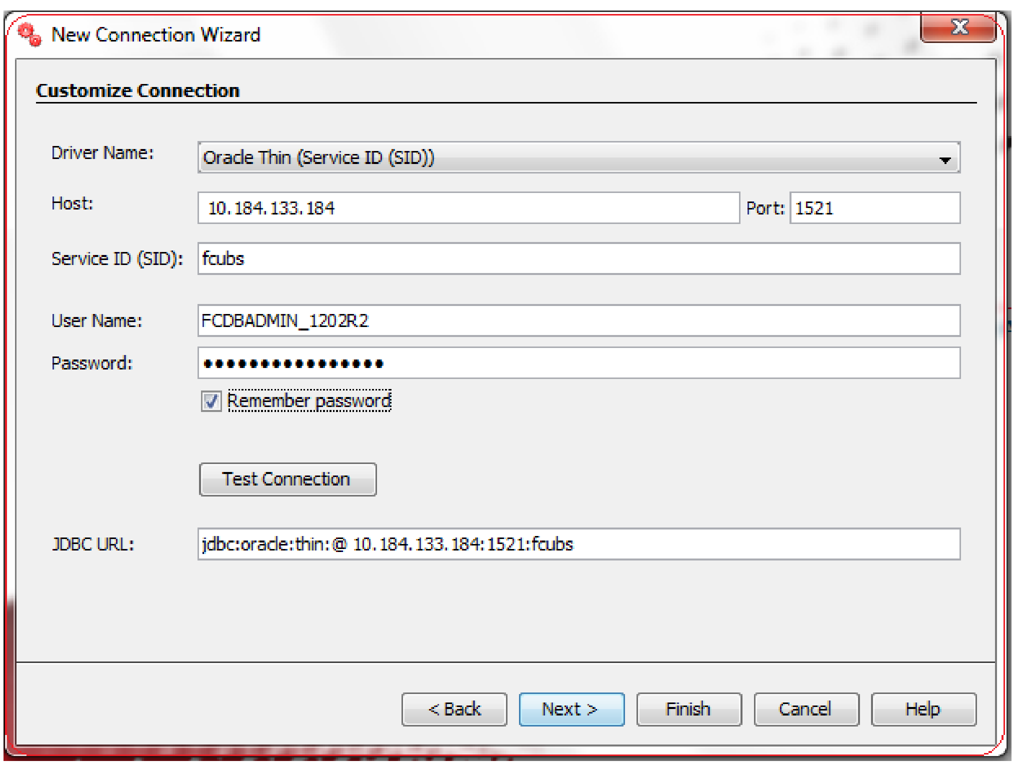
When you run the Development Workbench tool, you will get a pop up at home screen asking for the Database Connection as shown below:



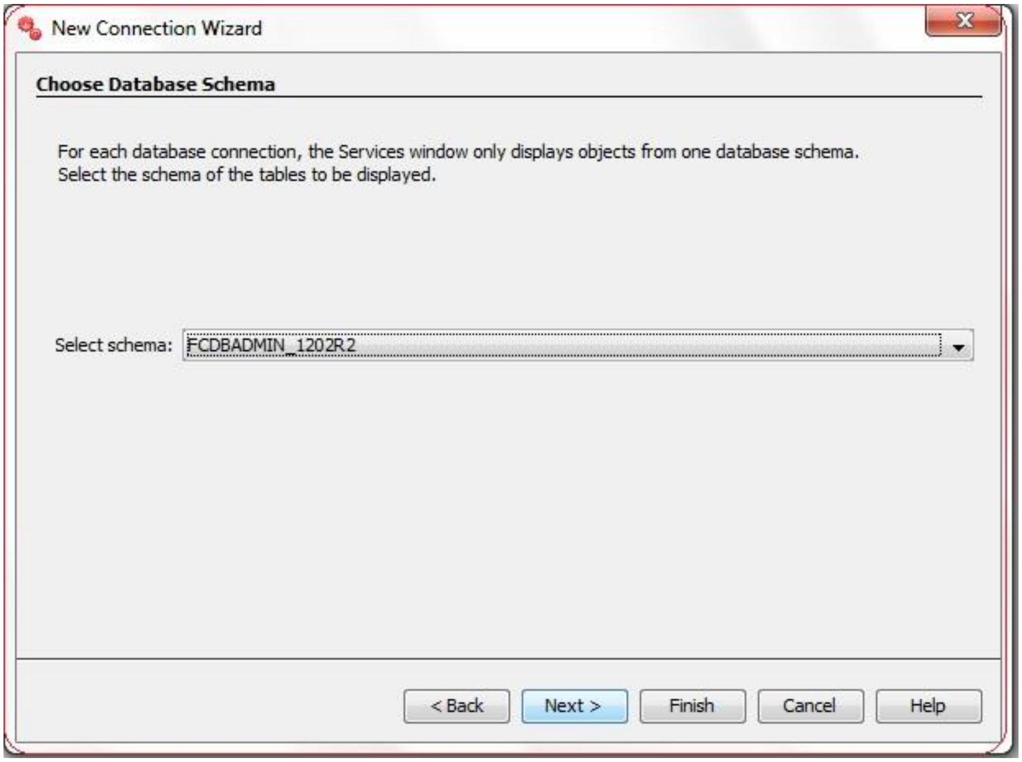
Select the connection and click on "OK". If no connection exists then Click on "New Database Connection", a wizard asking for the Driver details will appear as shown below.



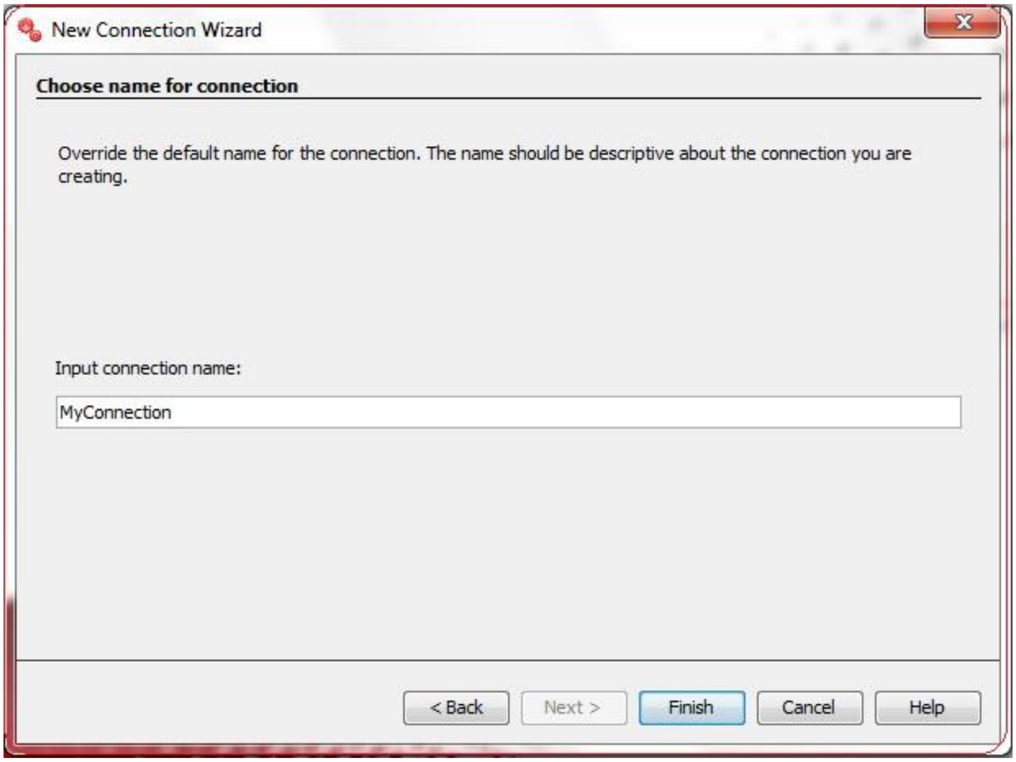
Enter the Driver details, and click on next button. Wizard next step will appear asking for the Database connection details as shown below.



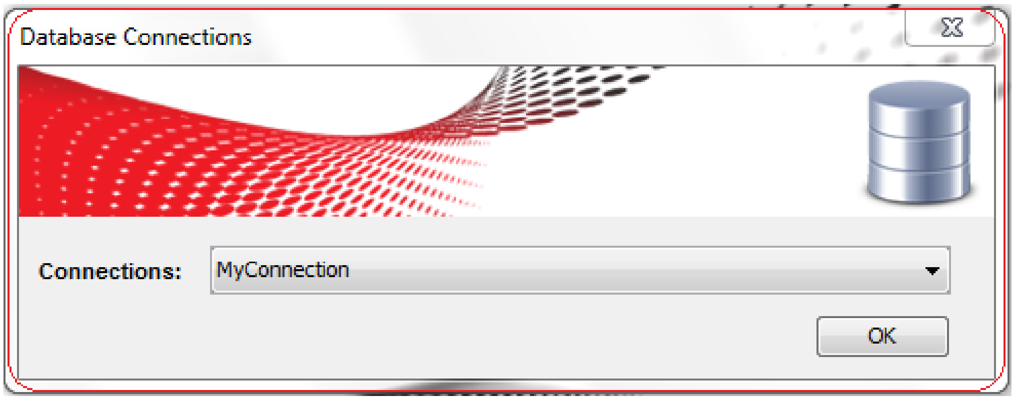
Enter the database connection details, and click on next button. Wizard next step will appear asking for database Schema as shown below.



Select the database schema, and click on next button. Wizard next step will appear asking for database alias name as shown below.



Enter the alias name for database, and Click on the Finish button user will be able to see the connection in the Database Connection Dialog as shown below.



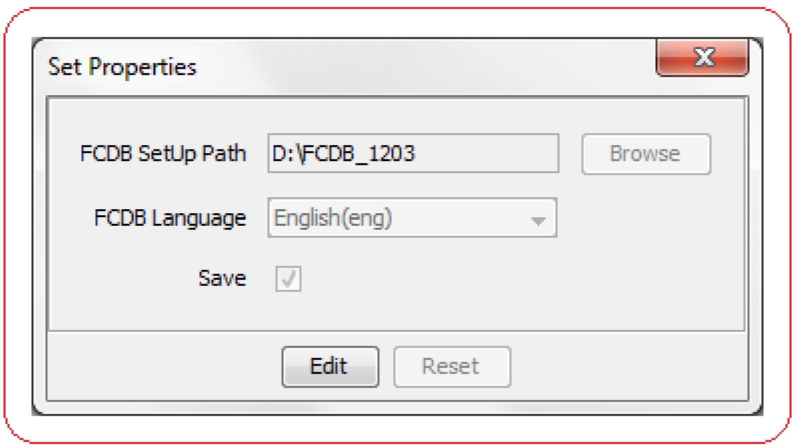
Click on Ok button to establish the connection to the database.

FCDB Properties

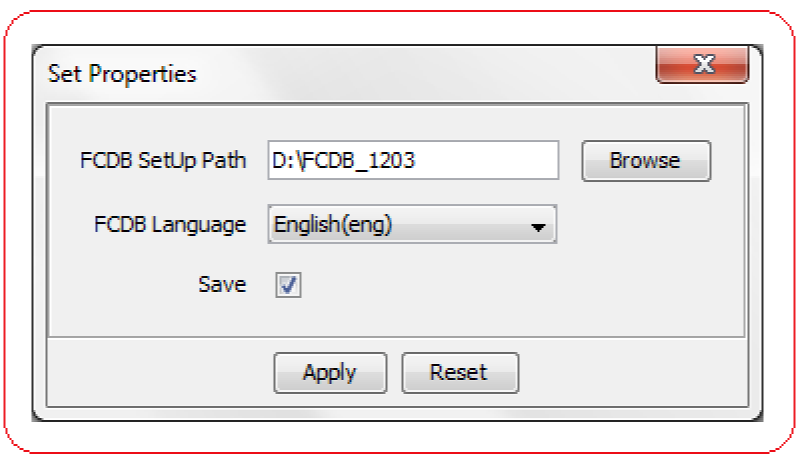
FCDB properties Dialog can be opened by clicking on the following highlighted icon in the tool bar.



FCDB Properties dialog will allow user to enter the path of the FCDB folder and the language. When you run the Development Workbench tool, you will get a pop up at home screen asking for the FCDB Properties as shown below:

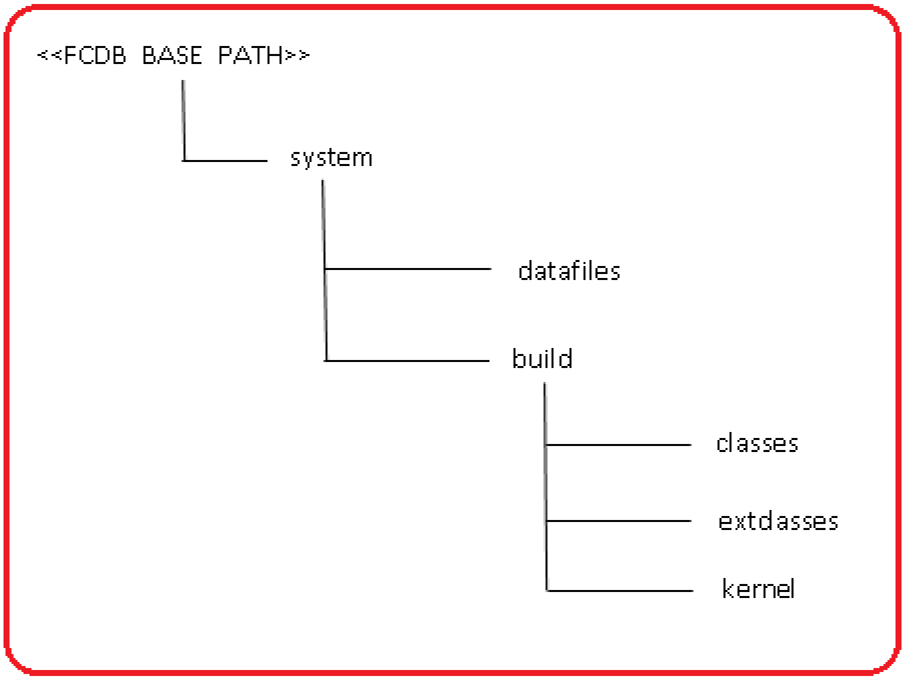


Click on the Edit button and user will be able to enter the details:

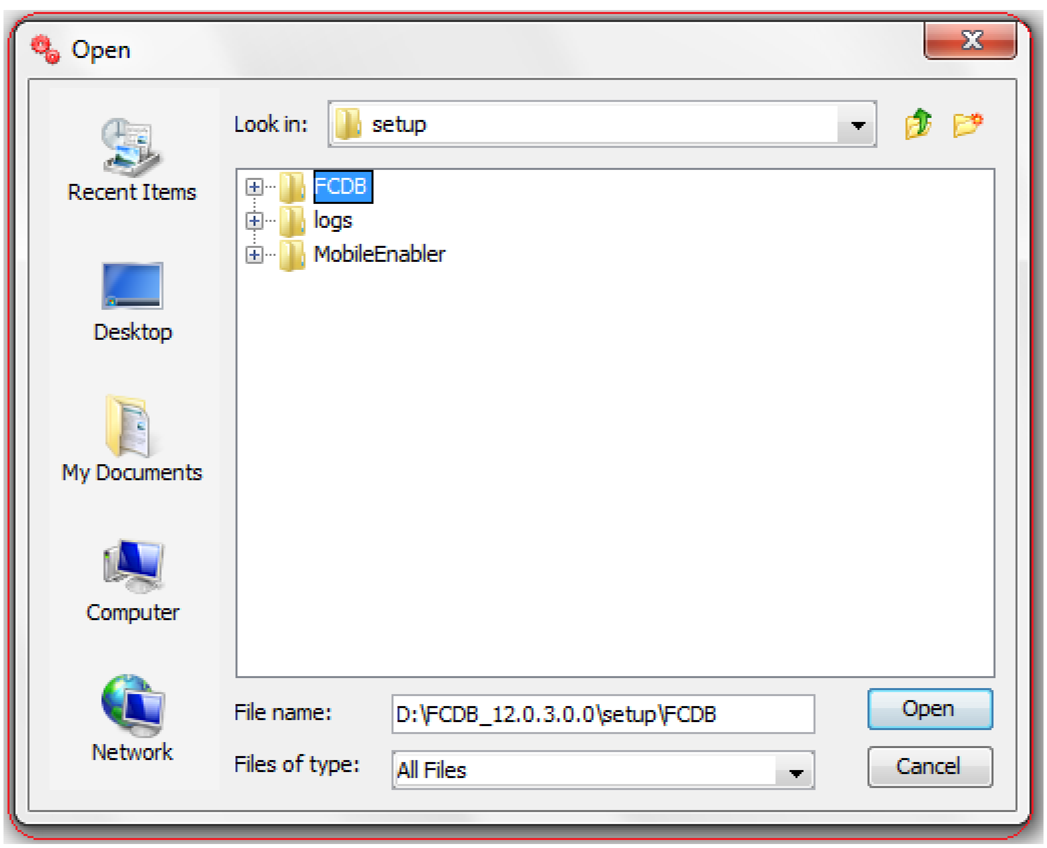


FCDB Setup Path

FCDB setup path will allow user to enter the path of FCDB base directory. This directory should contain FCDB_kernel.jar, third party libraries and datafiles in proper folder structure as shown below.

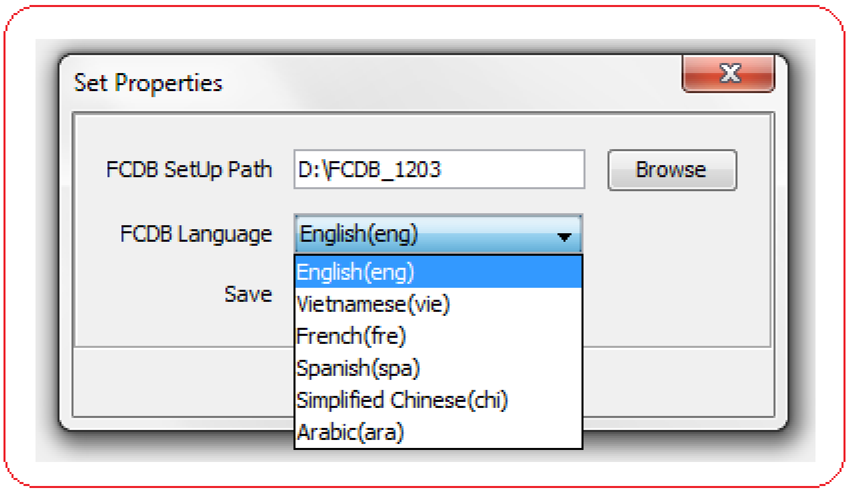


The FCDB home directory path can be browsed using browse button.

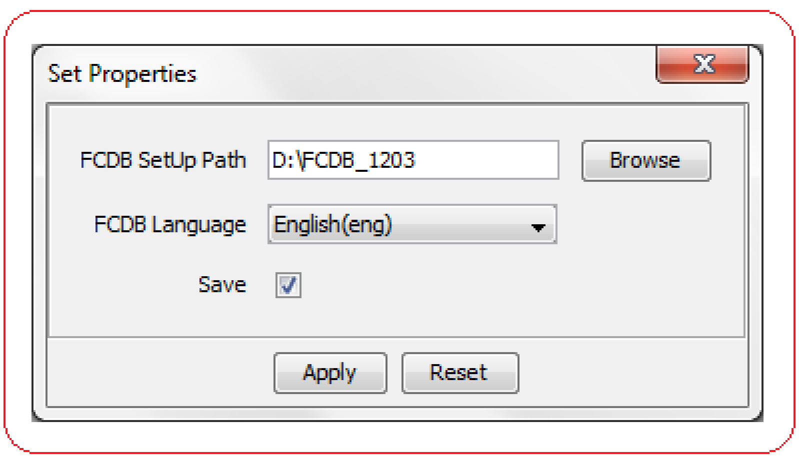


FCDB Language

FCDB language can be chosen from the available languages displayed in the drop down list. FCDB language is used in tool to preview the screen in that language and also load the language specific CSS.



Select any language from the drop down list, and click on apply button.



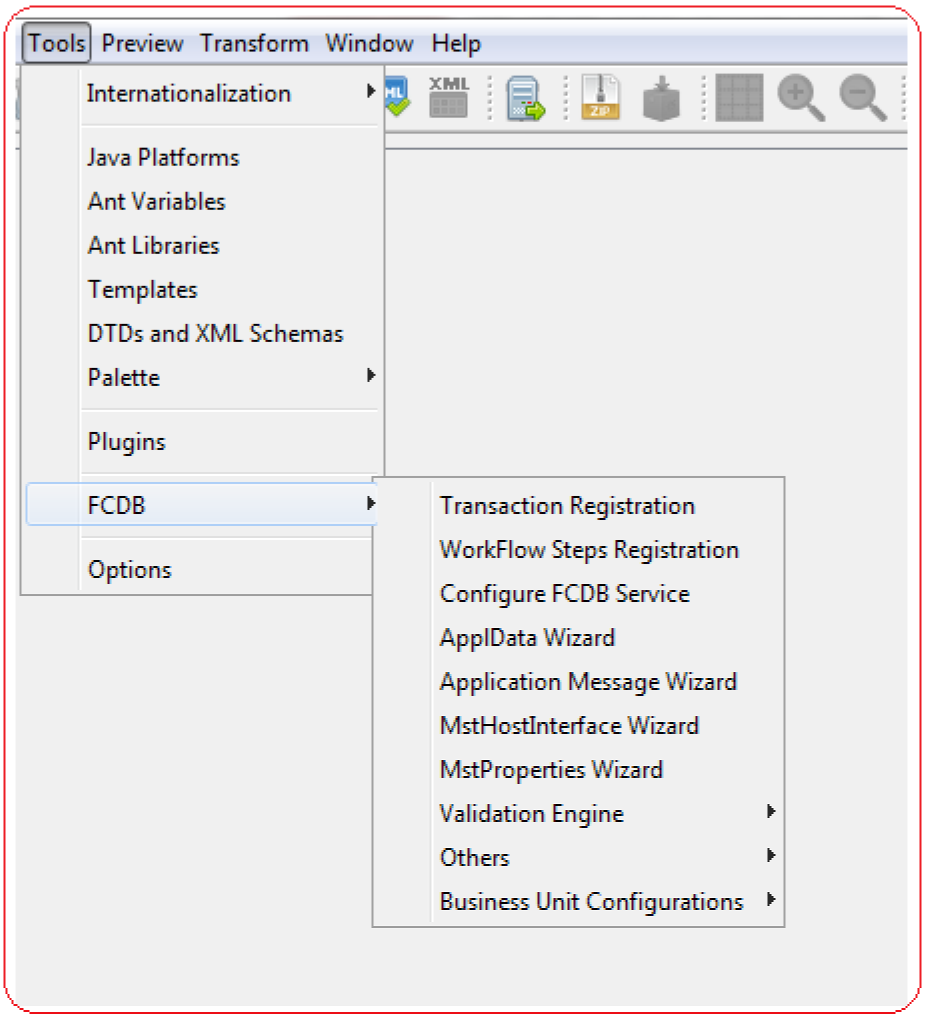
On clicking the Apply button the properties will be applied.

FCDB Application Configurations

The IDE provides the FCDB menu item which allows user to configure different FCDB configuration parameters. Configuration Wizards have been provided using which user can search for an existing configuration / Add new configuration in a step-wise manner. These wizards will generate the Add / Update / Delete scripts depending upon the action taken by the user and also allow user to execute them.

To use the FCDB Configuration Menu**,**

Go to **Tools** → **FCDB** as shown below:



The IDE allows user the following FCDB Configurations:

1. Transaction Registration

Transaction in FCDB can be defined as a business functionality which can be exposed to end user as a menu item. Transactions in FCDB are identified by three letter alphanumeric value, for e.g. OAT is transaction id of Own Account Transfer.

Transactions needs to be defined/registered in following two tables of FCDB application

I. **MstTxn**

This is the parent table where all FCDB transactions, irrespective of the fact that it is meant for a specific entity/user segment/channel, needs to be registered.

II. **MstUserTypeTxn**

This is child table where a developer can register the transactions for an entity/usertype/channel combination. Depending upon the entity/usertype/channel combination for which a transaction is registered bank administrator will get the transaction list while defining role for that entity/usertype/channel.

MstTxn Table Definition

Column Name	Type	Nullable	Default	Comments
IDTXN	CHAR(3 CHAR)	No		Three letter alphanumeric value to uniquely identify this transaction.
DESCRIPTION	VARCHAR2(255 CHAR)	Yes		Description of Transaction.
IDAPP	CHAR(2 CHAR)	No	'RR'	App ID ('RR' or 'A1').
TXNGROUP	CHAR(10 CHAR)	Yes	'*'	Group under which this transaction will come, description of group is defined in appldata table with dataname = 'TXN_GROUP_DESC' and datavalue = <Value of this column>.
IDSEQ	NUMBER	Yes		
GIFNAME	VARCHAR2(255 CHAR)	Yes		
ISMENUTXN	CHAR(1 CHAR)	No	'Y'	Flag to make this transaction available in user menu.
TOKEN1	VARCHAR2(255 CHAR)	Yes		
TOKEN2	VARCHAR2(255 CHAR)	Yes		Main menu under which this transaction will come, description of main menus is defined in appldata table with dataname = 'MENU_TAB_DESC' and datavalue = <Value of this column>.
TOKEN3	VARCHAR2(255 CHAR)	Yes		
TOKEN4	VARCHAR2(255 CHAR)	Yes		

TOKEN5	VARCHAR2(255 CHAR)	Yes		Numeric value to define the position of transaction in the transaction group.
LAYOUT_TYPE	VARCHAR2(5 CHAR)	Yes		
IDPROXYREQD	VARCHAR2(1 CHAR)	Yes	'Y'	
FLAGSERVICEREQUEST	CHAR(1 CHAR)	No	'N'	Flag to enable this transaction as a Service Request transaction.
ENABLETXNBLACKOUT	VARCHAR2(1 CHAR)	Yes	'N'	Flag to enable black out functionality for this transaction.
LIMITSALLOWED	VARCHAR2(1 CHAR)	No	'N'	Flag to enable limits functionality for this transaction.
ISEXTERNAL	VARCHAR2(1 CHAR)	Yes	'N'	
TYPECUST	VARCHAR2(20 CHAR)	Yes		Comma separated list of applicable Customer Types for this transaction. Possible values are C - Bank Customer, D - Card Customer, I- Investor Services Customer, if no value specified application will default the value to 'C'.
CODTXN	VARCHAR2(4 CHAR)	No	'OMTG'	Group under which this transaction will come in account mapping, description of group is defined in appldata table with dataname = 'CODTXN_DESC' and datavalue = <Value of this column >.
IDMODULE	VARCHAR2(3 CHAR)	Yes		Any specific Module to which this transaction belongs, helpful to get the list of registered transactions in the application specific to a module.
TYPETXN	CHAR(1 CHAR)	No	'N'	Type of transaction, Financial(F)/Non-Financial(N)/Inquiry(I).
AUTHPARAM	VARCHAR2(6 CHAR)	Yes	'111110'	This field indicates parameters required to manage authorization rules. It consists of 6 digit number (consisting of 0 & 1) where 0 resembles 'Not required' and 1 resembles 'Required'. Following configurations are driven by each character 1st Char - Branch, 2nd Char - Account, 3rd Char - Currency, 4th Char - From Amount, 5th Char - To Amount, 6th char - 0:Generate Error, 1:Generate warning, 2:Send for release.
ADTNL_PARAMS	VARCHAR2(2000 CHAR)	Yes		Additional Parameters for this Transaction.

REF_ACCTTYPE	VARCHAR2(10 CHAR)	Yes		Applicable Type of Accounts or combination of type of Accounts for this transaction, like CASA/TD/LOAN(C/T/L/CL/CLT).
PROXYTXNID	CHAR2(3 CHAR)	Yes		Proxy Transaction ID.
ALLOWED_PRIVILEGES	VARCHAR2(20 CHAR)	Yes		# separated list of possible privileges to be given to this transaction. Privileges are I (Initiation), A (Authorization) and V (View).
LIVEHELPMODULEID	VARCHAR2(10 CHAR)	Yes	'NON'	Live Help Module ID for this transaction, Modules are defined in LIVEHELPCONFIG table.

MstUserTypeTxn Table Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2(5 CHAR)	No		Entity ID for which this transaction needs to be registered.
USERTYPE	VARCHAR2(3 CHAR)	No		User Segment for which this transaction needs to be registered.
IDTXN	CHAR(3 CHAR)	No		Three letter alphanumeric value to uniquely identify this transaction.
IDCHANNEL	VARCHAR2(3 CHAR)	No		Channel ID for which this transaction needs to be registered.
TXNGROUP	VARCHAR2(10 CHAR)	Yes	'*'	Group under which this transaction will come, description of group is defined in appldata table with dataname = 'TXN_GROUP_DESC' and datavalue = <Value of this column>.
IDSEQ	NUMBER	Yes		
GIFNAME	VARCHAR2(255 CHAR)	Yes		
ISMENUTXN	CHAR(1 CHAR)	No	'Y'	Flag to make this transaction available in user menu.
TOKEN1	VARCHAR2(30 CHAR)	Yes		
TOKEN2	VARCHAR2(30 CHAR)	Yes		Main menu under which this transaction will come, description of main menus is defined in appldata table with dataname = 'MENU_TAB_DESC' and datavalue = <Value of this column>.
TOKEN3	VARCHAR2(30 CHAR)	Yes		
TOKEN4	VARCHAR2(30 CHAR)	Yes		
TOKEN5	VARCHAR2(30 CHAR)	Yes		Numeric value to define the position of transaction in the transaction group.

ISDEFAULTNAV	CHAR(1 CHAR)	Yes	'N'	Flag to enable quick and easy access to transaction post login, transaction will be available as hyperlink after successful login.
FLAGSERVICEREQUEST	CHAR(1 CHAR)	No	'N'	Flag to enable this transaction as a Service Request transaction.
ENABLETXNBLACKOUT	VARCHAR2(1 CHAR)	No	'N'	Flag to enable black out functionality for this transaction.
LIMITSALLOWED	VARCHAR2(1 CHAR)	No	'N'	Flag to enable limits functionality for this transaction.
TYPECUST	VARCHAR2(20 CHAR)	Yes		Comma separated list of applicable Customer Types for this transaction. Possible values are C - Bank Customer, D - Card Customer, I- Investor Services Customer, if no value specified application will default the value to 'C'.
CODTXN	VARCHAR2(3 CHAR)	Yes	'OMTG'	Group under which this transaction will come in account mapping, description of group is defined in appldata table with dataname = 'CODTXN_DESC' and datavalue = <Value of this column>.
IDMODULE	VARCHAR2(10 CHAR)	Yes		Any specific Module to which this transaction belongs, helpful to get the list of registered transactions in the application specific to a module.
REF_USERTYPE	VARCHAR2(225 CHAR)	Yes		Comma separated list of usertypes this transaction can access.
ISENABLED	CHAR(1 CHAR)	No	'Y'	Flag to enable/disable this transaction in application.
ISEXTERNAL	VARCHAR2(1 CHAR)	Yes	'N'	
USERAGENTREQID	VARCHAR2(200 CHAR)	Yes		
ALLOWED_PRIVILEGES	VARCHAR2(20 CHAR)	Yes		# separated list of possible privileges to be given to this transaction. Privileges are I (Initiation), A (Authorization) and V (View).
ISTOOL	CHAR(1 CHAR)	Yes	'N'	Flag to make this transaction available in Quick Tools Widget.
TOOLTYPE	CHAR(1 CHAR)	Yes	'N'	Set the value to 'Y' to make this transaction available in the list of tools and calculators in Login Screen.
INITAUTHID	VARCHAR2(20 CHAR)	Yes		Authorization Engine to be called for this transaction. List of Authorization Engines are registered in table mstinitauthtypes.
INITREQUESTID	CHAR(7 CHAR)	Yes		Registered work flow step id (refer to MstChannelATS table) getting called on click of this transaction from menu. If no value specified first step will be called.
AUTHALERTREQUIRED	CHAR(1 CHAR)	Yes	'N'	Flag to enable alert generation by Authorization Engine.

ALERTPARAM	VARCHAR2(4000 CHAR)	Yes		Parameters for Alert.
CUTOFFALLOWED	CHAR(1 CHAR)	No	'N'	Flag to enable Cut-Off functionality for this transaction.
QUICKTASK	CHAR(1 CHAR)	Yes	'N'	Flag to make this transaction available in Quick Tasks Widget.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these Transactions in the above tables.

To Open Transaction Registration Wizard, Go to Menu Options

Tools → FCDB → Transaction Registration

Transaction Registration – Search

This step is meant to allow the user to search for the existing Transactions available in tables given below of the connected Database.

MstTxn: Table containing unique ID identifying the transaction, description, menu and functional details of transaction.

MstUserTypeTxn: Child Table of MstTxn containing Transaction ID, Entity ID, User Type and Channel ID to group transactions based on these fields.

Select the table in which you want to search for the FCDB Transactions. The Following Parameters are provided to user to search for existing transactions.

MstTxn

- ✓ Transaction ID
- ✓ Description

MstUserTypeTxn

- ✓ Transaction ID
- ✓ Description
- ✓ Entity ID
- ✓ UserType
- ✓ Channel ID

Tick the parameters to search transactions with, select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available transactions in the selected table matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.

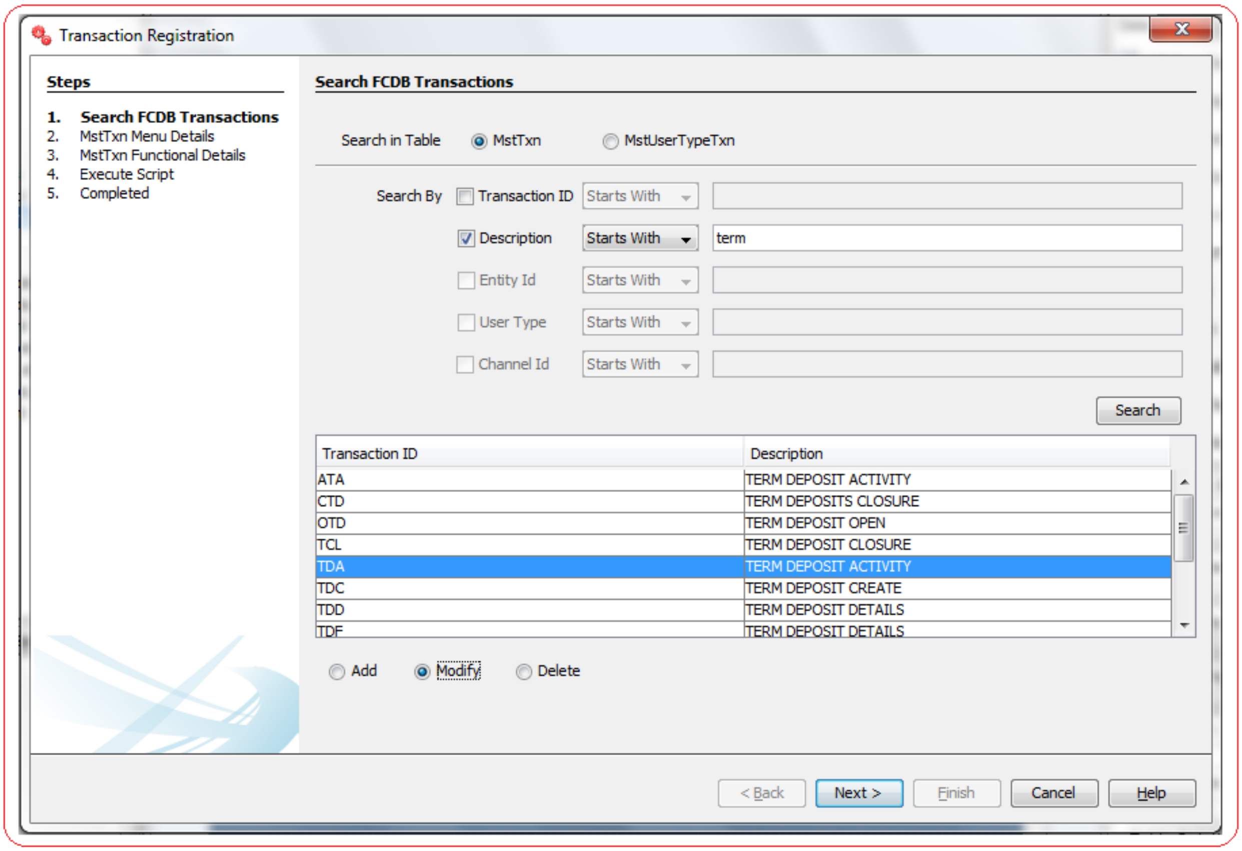
Depending upon the search output one can do the following operations,

I. **Add New:** Select this radio button to add new Transaction in the selected table.

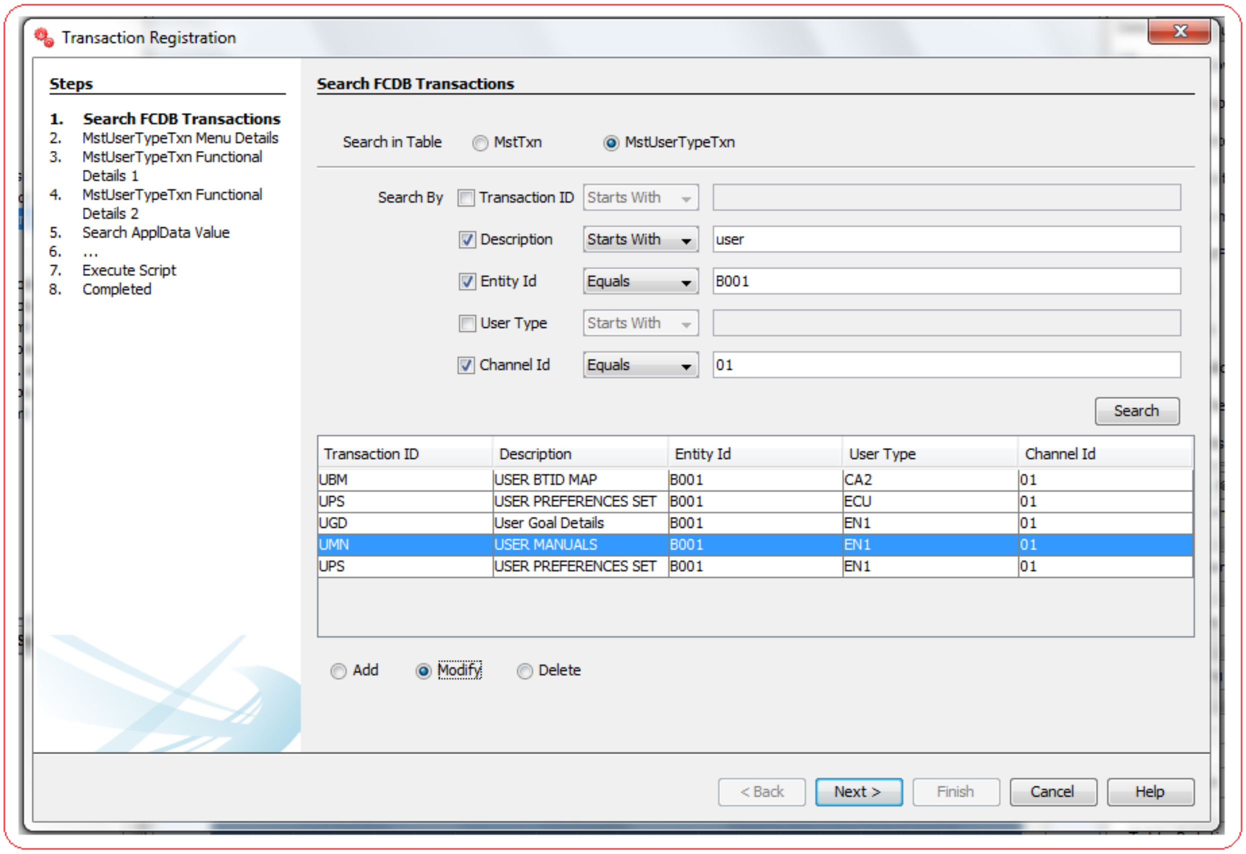
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.

III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

MstTxn - Search



MstUserTypeTxn - Search

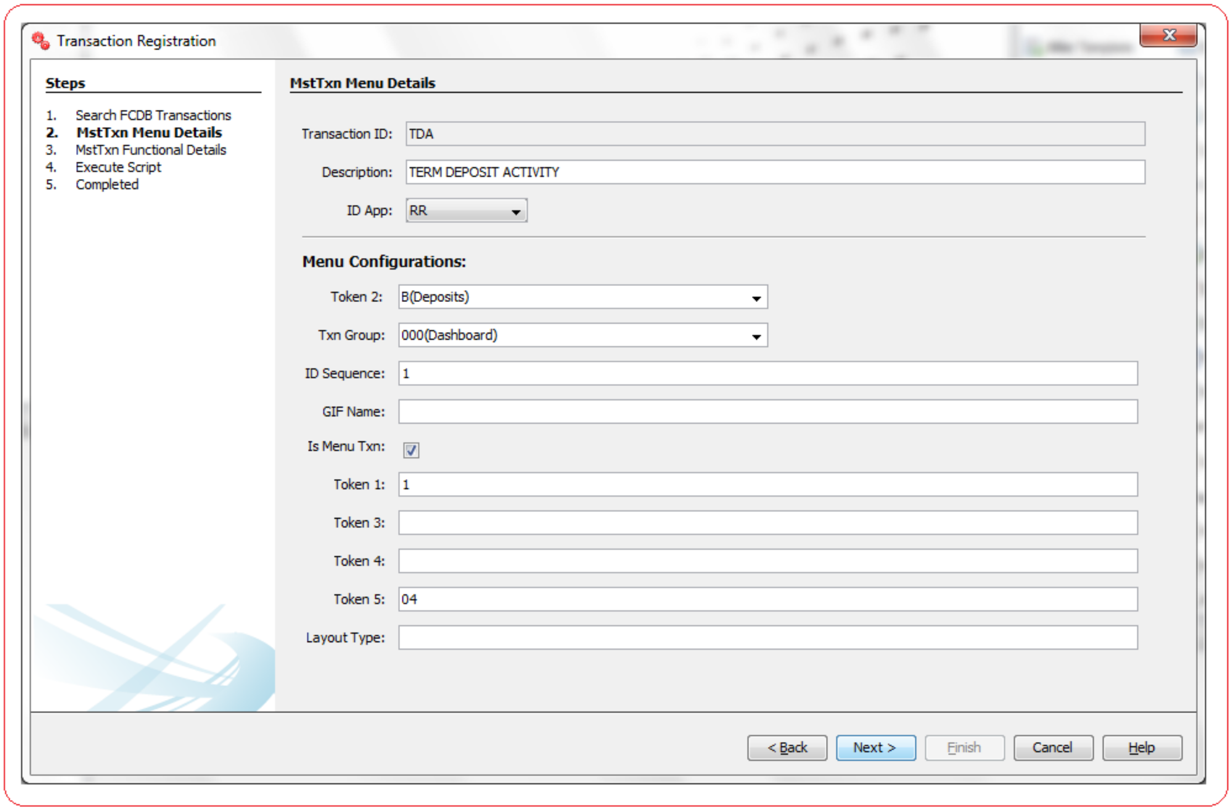


Transaction Registration - Edit

This step is meant to allow users to input details for a new Transaction record in MstTxn / MstUserTypeTxn, modify details of an existing one or view details of an existing one if user wants to delete the same.

I. MstTxn

A panel showing/asking for the Menu Details will be displayed. Enter the Menu Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstTxn Table.

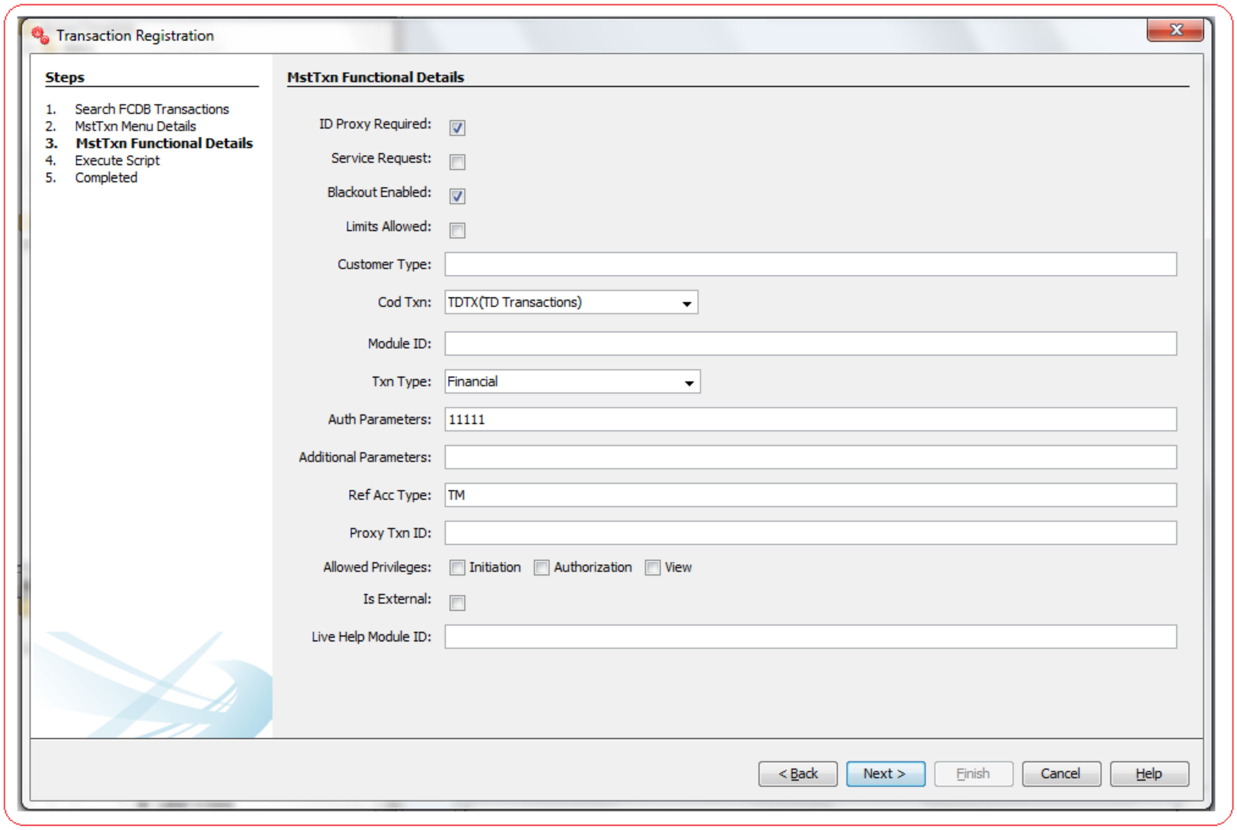


List of the User Input fields and their description are as follows:

- **Transaction ID:** Enter a three letter alphanumeric value to uniquely identify this transaction.
- **Description:** Enter the description of the transaction.
- **ID App:** Select the value from the drop down.
- **Token 2:** Select the main menu under which this transaction will come, description of main menus is defined in appldata table with dataname = 'MENU_TAB_DESC' and datavalue = <Value of this field>. To define a new main menu, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Txn Group:** Select the group under which this transaction will come, description of group is defined in appldata table with dataname = 'TXN_GROUP_DESC' and datavalue = <Value of this field>. To define a new group, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Is Menu Txn:** Flag to make this transaction available in user menu.. By default it is checked to show this Transaction in Menu. It should be unchecked for Service Request Transaction that may not be shown as menu item in FCDB application.
- **Token 5:** Enter a numeric value to define the position of transaction in the transaction group.

Click **Next** to go to next step.

A panel showing/asking for the Functional Details will be displayed. Enter the Functional Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstTxn Table.



List of the User Input fields and their description are as follows:

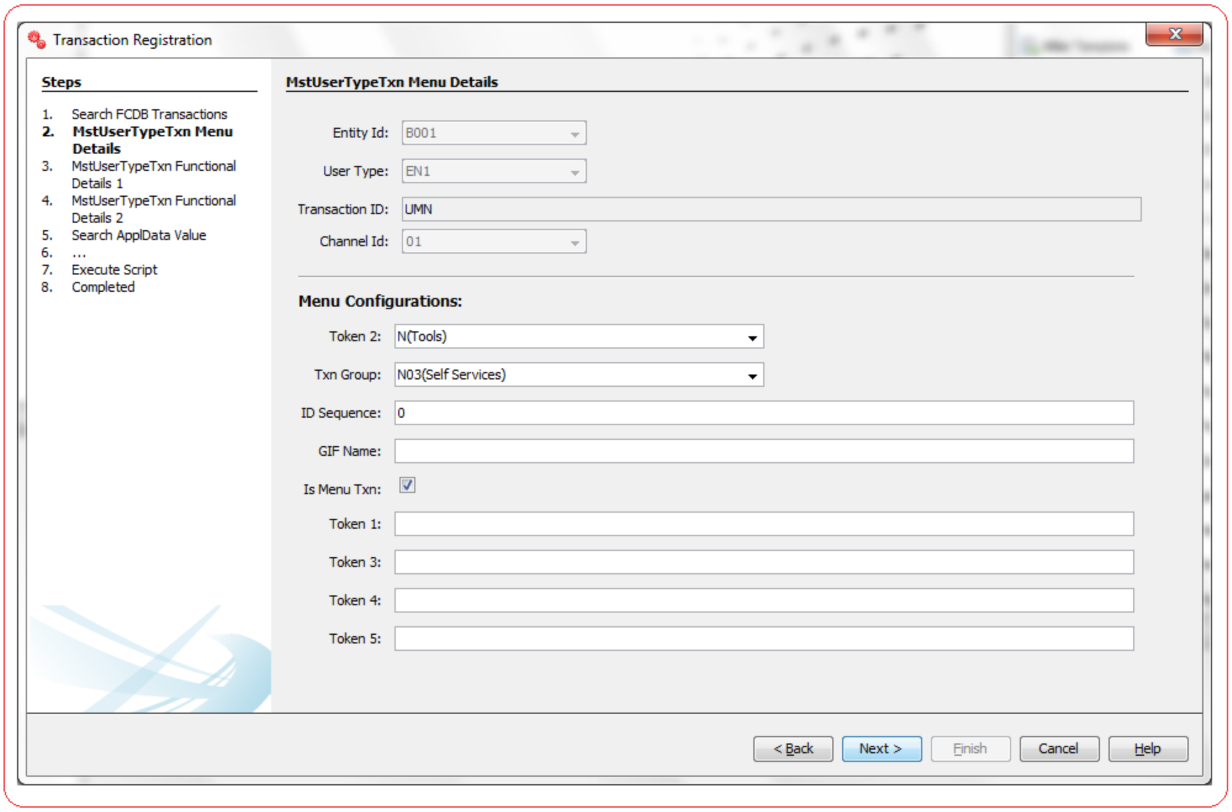
- **ID Proxy Required:** Flag to indicate if Proxy ID is required or not.
- **Service Request:** Flag to enable this transaction as a Service Request transaction.
- **Blackout Enabled:** Flag to enable black out functionality for this transaction.
- **Limits Allowed:** Flag to enable limits functionality for this transaction.
- **Customer Type:** Enter comma separated list of applicable Customer Types for this transaction. Possible values are C - Bank Customer, D - Card Customer, I- Investor Services Customer, if no value specified application will default the value to 'C'.
- **Cod Txn:** Group under which this transaction will come in account mapping, description of group is defined in appldata table with dataname = 'CODTXN_DESC' and datavalue = <Value of this field>. To define a new main menu, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Module ID:** Enter any specific Module to which this transaction belongs, helpful to get the list of registered transactions in the application specific to a module.
- **Txn Type:** Enter the type of transaction, Financial (F)/Non-Financial (N)/Inquiry (I).
- **Auth Parameters:** This field indicates parameters required to manage authorization rules. It consists of 6 digit number (consisting of 0 & 1) where 0 resembles 'Not required' and 1 resembles 'Required'. Following configurations are driven by each character 1st Char - Branch, 2nd Char - Account, 3rd Char - Currency, 4th Char - From Amount, 5th Char - To Amount, 6th char - 0:Generate Error, 1:Generate warning, 2:Send for release.
- **Additional Parameters:** Enter the Additional Parameters needed for this transaction.
- **Ref Acc Type:** Enter the Applicable Type of Accounts or combination of type of Accounts for this transaction, like CASA/TD/LOAN(C/T/L/CL/CLT).
- **Proxy Txn ID:** Enter the Proxy Transaction ID.
- **Allowed Privileges:** Select the allowed privileges to be given to this transaction.

- **Is External:**
- **Live Help Module ID:** Enter the Live Help Module ID to configure a “Live Help Module” for this transaction. The ID string should be any alpha-numeric. Further, after configuring the “Live Help Module” for transactions, each module would need to be configured for integration with the Live Help On Demand service. The database table LIVEHELPCONFIG holds the detailed configuration of such mapping.

Click **Next** to go to next step.

II. <u>MstUserTypeTxn</u>

A panel showing/asking for the Menu Details will be displayed. Enter the Menu Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstUserTypeTxn Table.



List of the User Input fields and their description are as follows:

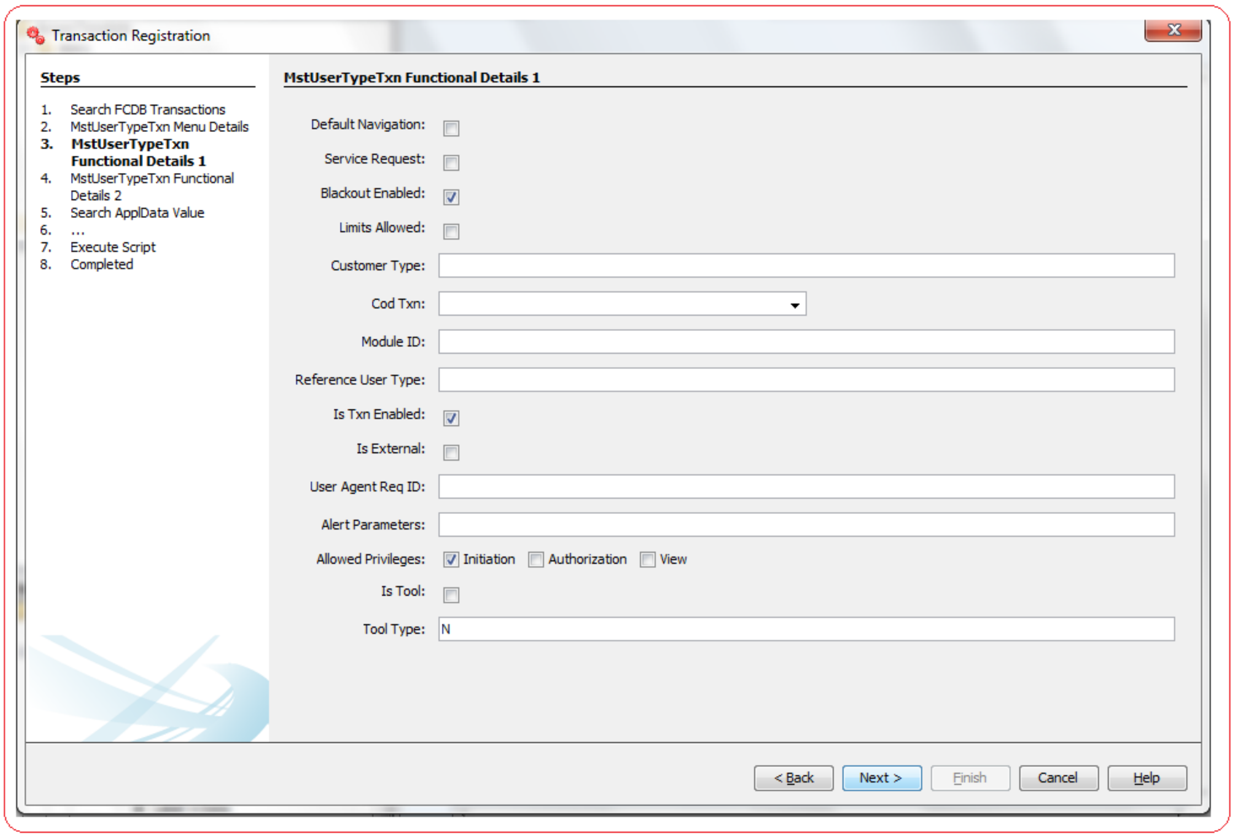
- **Entity ID:** Select the Entity ID for which this transaction needs to be registered.
- **User Type:** Select the User Segment for which this transaction needs to be registered.
- **Transaction ID:** Enter a three letter alphanumeric value to uniquely identify this transaction.
- **Channel ID:** Select the Channel ID for which this transaction needs to be registered.
- **Token 2:** Select the main menu under which this transaction will come, description of main menus is defined in appldata table with dataname = 'MENU_TAB_DESC' and datavalue = <Value

of this field>. To define a new main menu, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.

- **Txn Group:** Select the group under which this transaction will come, description of group is defined in appldata table with dataname = 'TXN_GROUP_DESC' and datavalue = <Value of this field>. To define a new group, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Is Menu Txn:** Flag to make this transaction available in user menu.. By default it is checked to show this Transaction in Menu. It should be unchecked for Service Request Transaction that may not be shown as menu item in FCDB application.
- **Token 5:** Enter a numeric value to define the position of transaction in the transaction group.

Click **Next** to go to next step.

A panel showing/asking for the Menu Details will be displayed. Enter the Menu Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstUserTypeTxn Table.



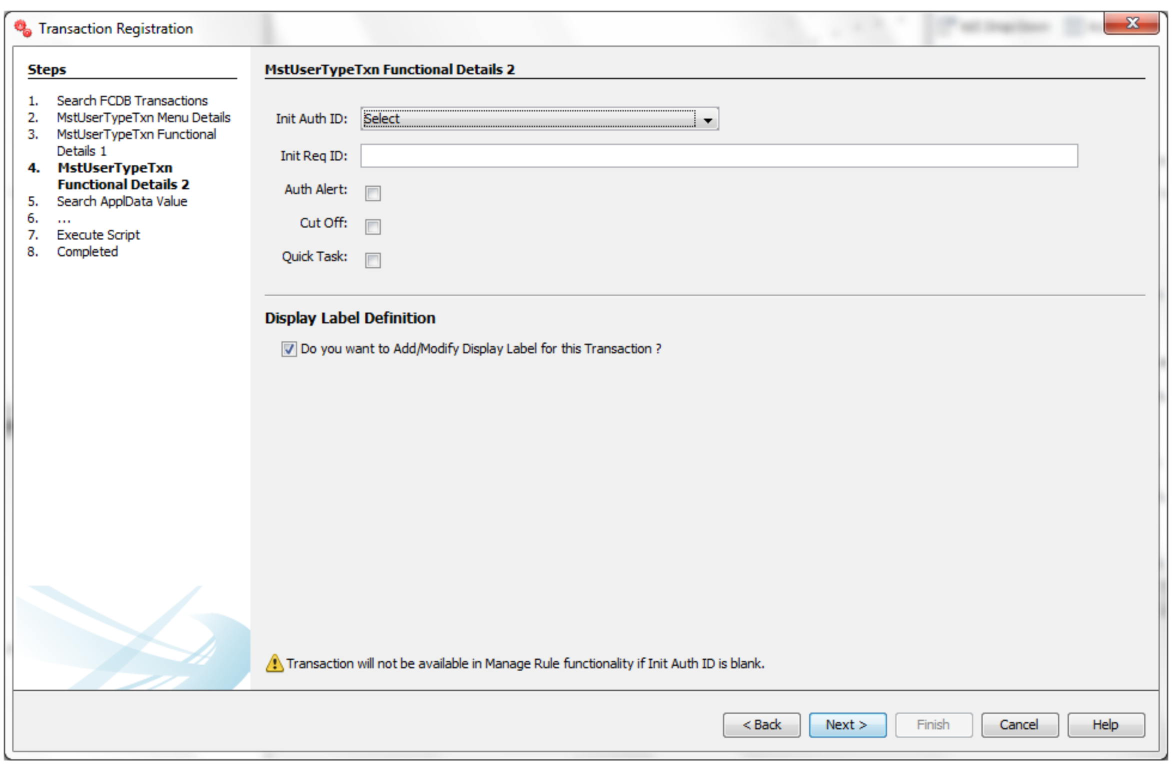
List of the User Input fields and their description are as follows:

- **Default Navigation:** Flag to enable quick and easy access to transaction post login, transaction will be available as hyperlink after successful login.
- **Service Request:** Flag to enable this transaction as a Service Request transaction.
- **Blackout Enabled:** Flag to enable black out functionality for this transaction.
- **Limits Allowed**: Flag to enable limits functionality for this transaction.

- **Customer Type:** Enter comma separated list of applicable Customer Types for this transaction. Possible values are C - Bank Customer, D - Card Customer, I- Investor Services Customer, if no value specified application will default the value to 'C'.
- **Cod Txn:** Group under which this transaction will come in account mapping, description of group is defined in appldata table with dataname = 'CODTXN_DESC' and datavalue = <Value of this field>. To define a new main menu, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Module ID:** Enter any specific Module to which this transaction belongs, helpful to get the list of registered transactions in the application specific to a module.
- **Reference User Type:** Enter the Comma separated list of usertypes this transaction can access.
- **Is Txn Enabled:** Flag to enable/disable this transaction in application. By default, it is checked to enable the transaction. A Transaction not enabled shall not get reflected in FCDB application and won't be available to user.
- **Is External:**
- **User Agent Req ID:**
- **Alert Parameters:** Enter the Alert Parameters needed for this transaction.
- **Allowed Privileges:** Select the allowed privileges to be given to this transaction.
- **Is Tool:** Flag to make this transaction available in Quick Tools Widget.
- **Tool Type:** Set the value to 'Y' to make this transaction available in the list of tools and calculators in Login Screen.

Click **Next** to go to next step.

A panel showing/asking for the Menu Details will be displayed. Enter the Menu Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstUserTypeTxn Table.



List of the User Input fields and their description are as follows:

- **Init Auth ID:** Enter the Authorization Engine to be called for this transaction. List of Authorization Engines are registered in table mstinitauthtypes.
- **Init Req ID:** Enter the Registered work flow step id (refer to MstChannelATS table) getting called on click of this transaction from menu. If no value specified first step will be called.
- **Auth Alert:** Flag to indicate if Authentication Alerts enabled for this transaction or not.
- **Cut OFF:** Flag to enable Cut-Off functionality for this transaction.
- **Quick Task:** Flag to make this transaction available in Quick Tasks Widget.

Click **Next** to go to next step.

2. WorkFlow Steps Registration

Oracle FLEXCUBE Direct Banking Application houses the table **MstChannelATS** where registration of transaction workflow steps needs to be done. For e.g. if Own Account Transfer (Txn ID: - OAT) comprises of three steps Initiate → Verify → Confirm, all these three steps needs to be registered in this table.

MstChannelATS Definition

Column Name	Type	Nullable	Default	Comments
IDREQUEST	VARCHAR2(50 CHAR)	N		Identifier for each step in the workflow. Nomenclature followed is "RR"+<Txn ID>+<Screen Sequence Number>. For e.g. for Own Account Transfer first step identifier will be **RROAT01.**
TYPPLUGIN	CHAR(1 CHAR)	N	'C'	Plugin Type ('T','C','E','X') to be used, each plugin type is mapped to an interfacing class. Preferred value 'C' for invoking FCDB Business tier.
AUDITREQUIRED	CHAR(1 CHAR)	N	'Y'	Flag to enable auditing of this work flow step in FCDB application.
IDCHANNEL	CHAR(2 CHAR)	N		Channel for which entry is being done, for e.g. 01 for Internet.
IDTXN	CHAR(3 CHAR)	N		Registered ID of the transaction, for e.g. OAT.
REQUIRESLOGIN	CHAR(1 CHAR)	N	'Y'	Flag to set if user login into FCDB application is mandatory to use this transaction work flow step.
CONTENTSTYLE	VARCHAR2(20 CHAR)	N	'HTML'	Style ID to be used to generate the content, Each style id is mapped to respective content generator java class in <identity>.xml available in home folder of FCDB base directory.
NAMRESOURCE	VARCHAR2(50 CHAR)	N	'dummy.xsl'	Name of the user interface XSL file to generate the desired UI content.
IDSERVICE	VARCHAR2(100 CHAR)	N		Name of the service XSL file containing the name of FCDB Business service to be invoked for this work flow step.
IDAPP	CHAR(2 CHAR)	N	'RR'	
FLAGPREPROCESS	CHAR(1 CHAR)	Y		Predefined processing features for data storage in **USERSESSIONDATA** table which can be used to provide simple data-security during a transactional workflow. Depending upon the value

				processing is applied on the request path before generating the service request XML.
FLAGPOSTPROCESS	CHAR(1 CHAR)	Y		Predefined processing features for data storage in **USERSESSIONDATA** table which can be used to provide simple data-security during a transactional workflow. Depending upon the value processing is applied on the response path before response is passed to Presentation tier.
NAMEOTRESOURCE	VARCHAR2(50 CHAR)	N		Name of the user interface XSL file to generate the desired UI content in case of any error leading to end of transaction (EOT).
NAMEOSRESOURCE	VARCHAR2(50 CHAR)	N		Name of the user interface XSL file to generate the desired UI content in case of any error leading to end of session (EOS).
ISONLYVALIDATEREQUEST	CHAR(1 CHAR)	N	'N'	Flag to validate the request data without invoking the actual business service, usually configured to 'Y' for verification steps in workflow.
IDSERVICE_EOT	VARCHAR2(50 CHAR)	Y		Name of the service XSL file containing the name of FCDB Business service to be invoked in case of any error leading to end of transaction (EOT).
IDSERVICE_EOS	VARCHAR2(50 CHAR)	Y		Name of the service XSL file containing the name of FCDB Business service to be invoked in case of any error leading to end of session (EOS).
IDREQUEST_EOT	VARCHAR2(50 CHAR)	Y		Identifier for step in the workflow to be called in case of any error leading to end of transaction (EOT). Nomenclature followed is "RR"+<Txn ID>+<Screen Sequence Number>.
AUTHREQUIRED	CHAR(1 CHAR)	N	'N'	Flag to enable the work flow step to pass through FCDB authorization engine.
TXNPWDREQUIRED	CHAR(1 CHAR)	N	'N'	Flag to enable transaction password screen for this workflow step.
FLAGPAGINATION	NUMBER(1)	Y		Set the value to 1 to call FCDB pagination infrastructure to

				display the data in a tabular grid supporting pagination.
ADTNL_PARAMS	VARCHAR2(4000 CHAR)	Y		
FLGORCH	CHAR(1 CHAR)	N	'C'	Flag to enable external orchestration, value 'C' to set no external orchestration.
ISONLYAUTHVALIDATE	CHAR(1 CHAR)	N	'N'	Flag to enable only module validations for this work flow step in FCDB authorization engine. FCDB Authorization Engine doesn't generate any reference number, usually configured to 'Y' for verification steps in workflow.
ISGENERICTEMPLATE	VARCHAR2(1 CHAR)	Y	'N'	This flag to be set to 'Y' if this workflow step will call GenericPaymentsService.
HASEXTENDEDRESPONSE	CHAR(1 CHAR)	Y	'Y'	Flag to set if extended response handler class needs to process the data for this workflow step.
ALERTREQUIRED	CHAR(1 CHAR)	Y	'N'	Flag to set if alert message needs to be generated.
VALID_IDREQUEST	VARCHAR2(100 CHAR)	Y		Comma separated valid idrequest(s) that can be source to this idrequest. Null: indicates any idrequest as source.
TYPE_REQUEST	CHAR(1 CHAR)	Y	'T'	'T' or 'W' (Transaction or Widget)
AUDITRESOURCEXSL	VARCHAR2(50 CHAR)	Y		Name of the user interface XSL file to generate the desired UI content for audit view, used in special cases where audit data is not HTML.
AUDITCONTENTSTYLE	VARCHAR2(20 CHAR)	Y		Style ID to be used to generate the audit data content, used in special cases where audit data is not HTML. Each style id is mapped to respective content generator java class in <identity>.xml available in home folder of FCDB base directory.

To Open MstChannelATS Wizard, Go to Menu Options
Tools → FCDB → WorkFlow Steps Registration

MstChannelATS – Search

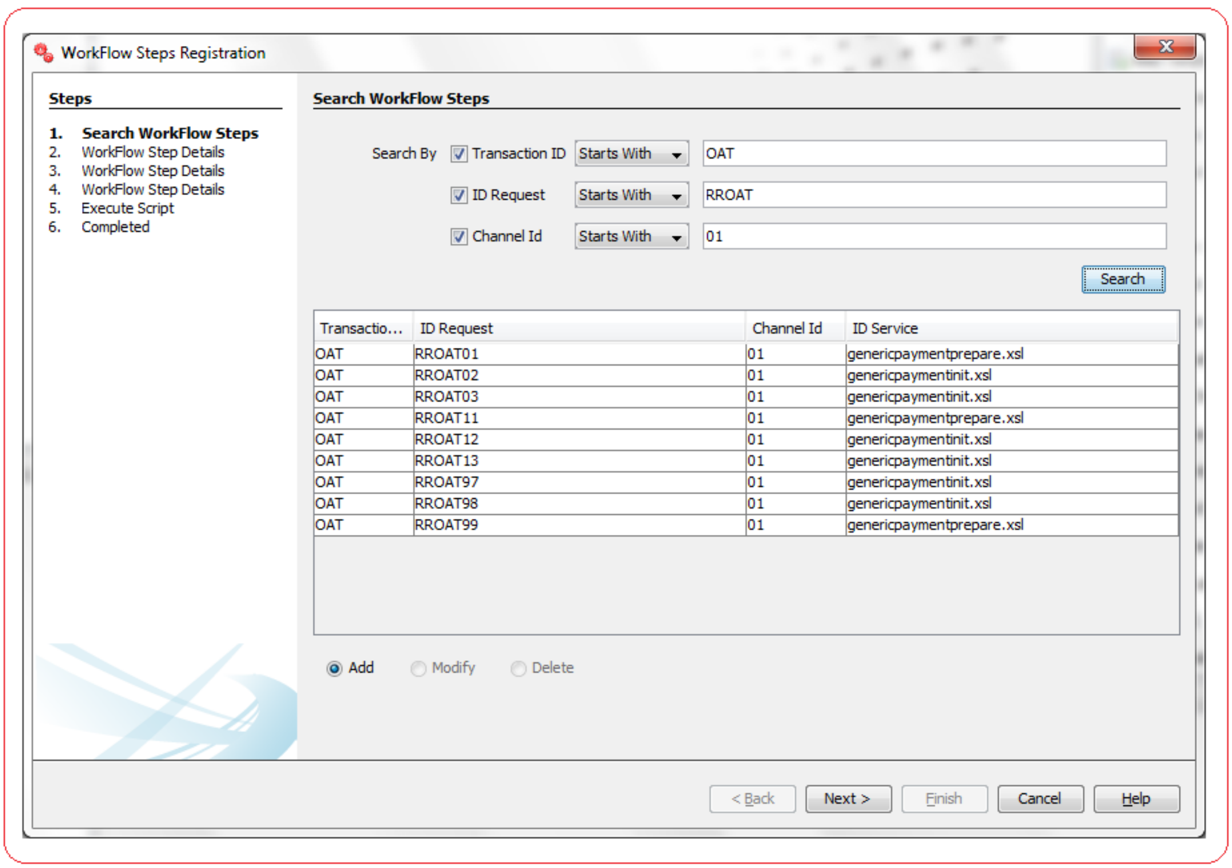
This step is meant to allow the user to search for the existing MstChannelATS configurations available in **MstChannelATS** table if connected to the Database.

A three criteria search is provided to user to search for existing MstChannelATS values. User can use any of the following parameters to search for existing entries.

- ✓ Transaction Id
- ✓ Id Request
- ✓ Channel Id

Select any of the above parameters, enter the search text and click on "Search" Button. The list of available results as per the search criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.

It is always advisable to search for existing entries before adding a new value in this table.



Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add a new value in MstChannelATS table.

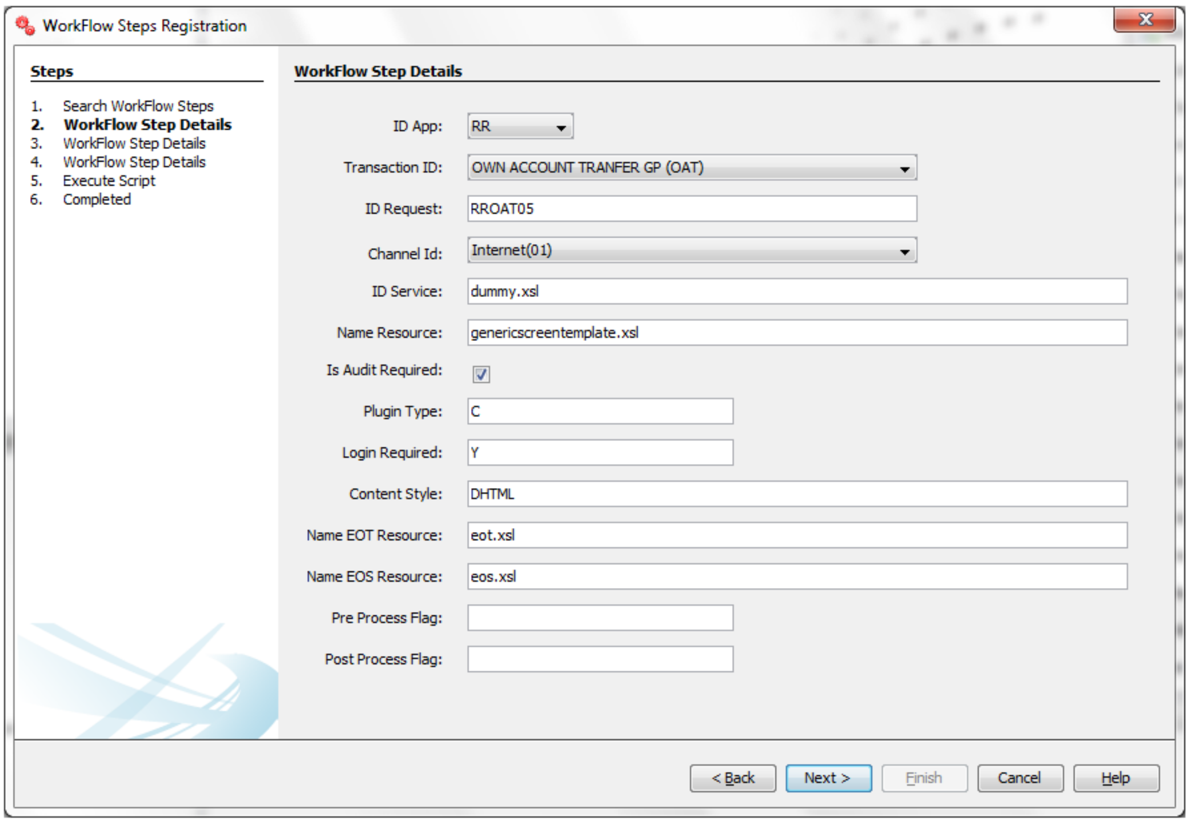
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected MstChannelATS value.

III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected MstChannelATS entry from the MstChannelATS table.

Click **Next** to go to next step.

MstChannelATS - Edit

This step is meant to allow users to input details for a new MstChannelATS entry, modify details of an existing one or view details of an existing one if user wants to delete the same.



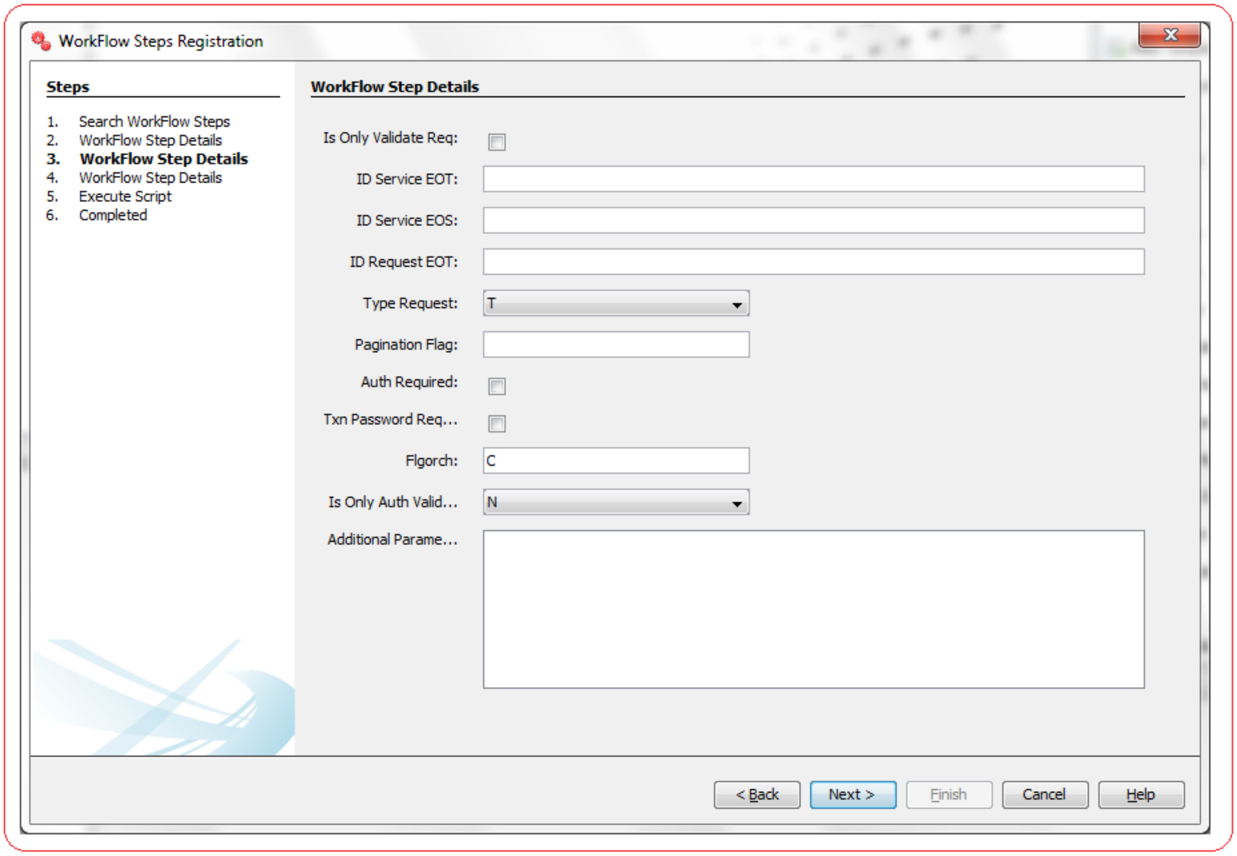
List of the User Input fields and their description are as follows:

- **ID App:** Select the value from the drop down.
- **Transaction ID:** Enter the Registered ID of the transaction, for e.g. OAT.

- **ID Request:** Enter the Identifier for this step in the workflow. Nomenclature followed is "RR"+<Txn ID>+<Screen Sequence Number>. For e.g. for Own Account Transfer first step identifier will be**RROAT01**.
- **Channel Id:** Select the Channel for which entry is being done, for e.g. 01 for Internet.
- **ID Service:** Name of the service XSL file containing the name of FCDB Business service to be invoked for this work flow step.
- **Name Resource:** Name of the user interface XSL file to generate the desired UI content.
- **Is Audit Required:** Flag to enable auditing of this work flow step in FCDB application.
- **Plugin Type:** Plugin Type ('T','C','E','X') to be used, each plugin type is mapped to an interfacing class. Preferred value 'C' for invoking FCDB Business tier.
- **Login Required:** Flag to set if user login into FCDB application is mandatory to use this transaction work flow step.
- **Content Style:** Style ID to be used to generate the content, Each style id is mapped to respective content generator java class in <identity>.xml available in home folder of FCDB base directory. For example: - Html, DHTML etc.
    - ✓ DHTML: Dynamic HTML is used when screen is generated using Development Workbench framework having NAMRESOURCE='genericscreentemplate.xsl'.
    - ✓ HTML: It is used when UI is designed manually by creating a .xsl file.
- **Name EOT Resource:** Name of the user interface XSL file to generate the desired UI content in case of any error leading to end of transaction (EOT).
- **Name EOS Resource:** Name of the user interface XSL file to generate the desired UI content in case of any error leading to end of session (EOS).
- **Pre Process Flag:** Enter the value in this field when you want to store your request data into USERSESSIONDATA table before doing the process. The Sample value for this field is 1,2,3,4.
    - ✓ Value “1” indicates FCDB will select the request data from USERSESSIONDATA table.
    - ✓ Value “2” indicates FCDB will store the request data into USERSESSIONDATA table.
    - ✓ Value “3” indicates FCDB will delete the request data from USERSESSIONDATA table.
    - ✓ Value “4” indicates FCDB will select and update the request data from USERSESSIONDATA table.
- **Post Process Flag:** Enter the value in this field when you want to store your response data into USERSESSIONDATA table before doing the process. The Sample values for this field are 1,2,3,4.
    - ✓ Value “1” indicates FCDB will select the response data from USERSESSIONDATA table.
    - ✓ Value “2” indicates FCDB will store the response data into USERSESSIONDATA table.
    - ✓ Value “3” indicates FCDB will delete the response data from USERSESSIONDATA table.
    - ✓ Value “4” indicates FCDB will select and update the response data from USERSESSIONDATA table.

Click **Next** to go to next step.

A panel showing / asking for the details will be displayed. Enter the Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstChannelATS Table.

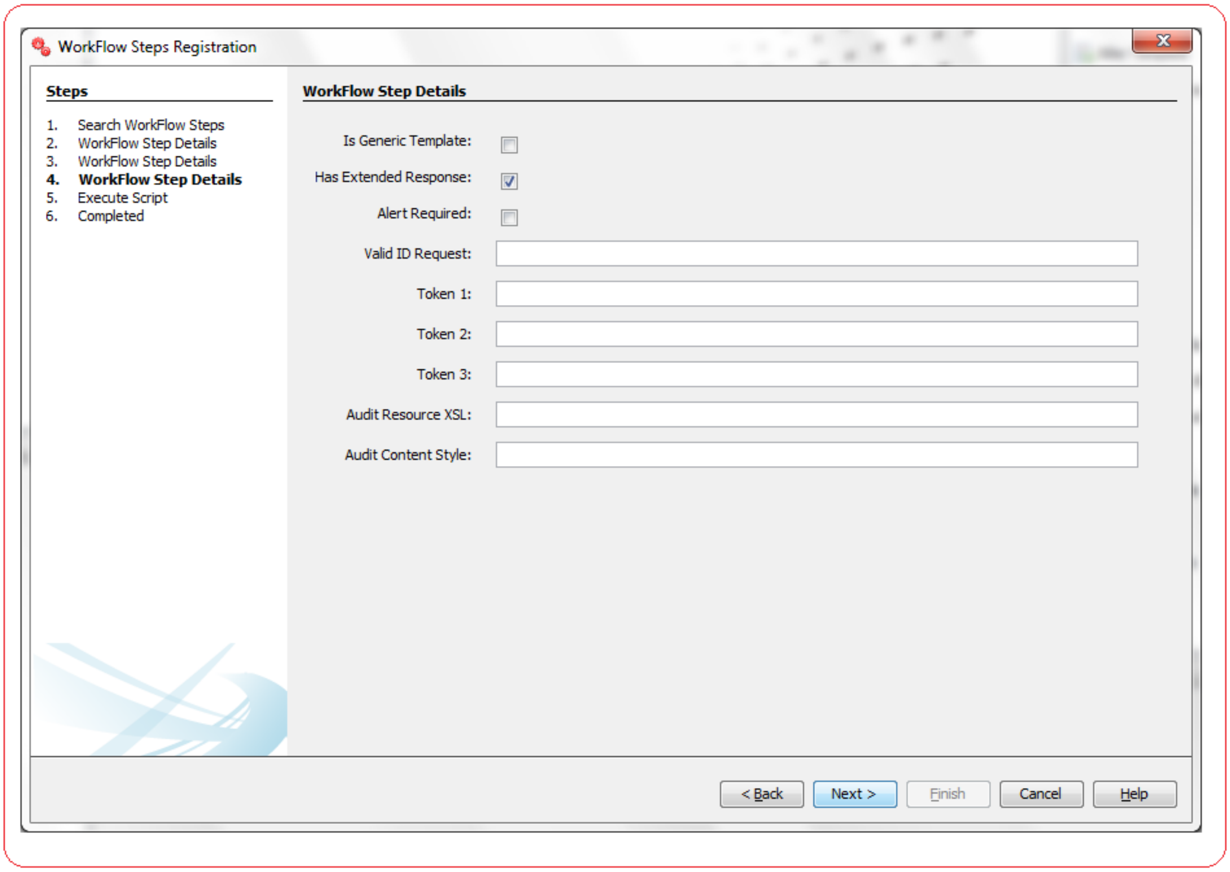


List of the User Input fields and their description are as follows:

- **Is Only Validate Req:** Flag to validate the request data without invoking the actual business service, usually configured to 'Y' for verification steps in workflow.
- **ID Service EOT:** Enter name of the service XSL file containing the name of FCDB Business service to be invoked in case of any error leading to end of transaction (EOT).
- **ID Service EOS:** Enter name of the service XSL file containing the name of FCDB Business service to be invoked in case of any error leading to end of session (EOS).
- **ID Request EOT:** Enter the Identifier for step in the workflow to be called in case of any error leading to end of transaction (EOT). Nomenclature followed is "RR"+<Txn ID>+<Screen Sequence Number>.
- **Type Request:** Type Request field is used to identify whether this is a transaction request or widget request ('T' or 'W').
- **Pagination Flag:** Set the value to 1 to call FCDB pagination infrastructure to display the data in a tabular grid supporting pagination.
- **Auth Required:** Flag to enable the work flow step to pass through FCDB authorization engine.
- **Txn Password Required:** Flag to enable transaction password screen for this workflow step.
- **Flgorch:** Flag to enable external orchestration, value 'C' to set no external orchestration.
- **Is Only Auth Validate:** Flag to enable only module validations for this work flow step in FCDB authorization engine. FCDB Authorization Engine doesnot generate any reference number, usually configured to 'Y' for verification steps in workflow.

Click **Next** to go to next step.

A panel showing/asking for the details will be displayed. Enter the Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the MstChannelATS Table.



List of the User Input fields and their description are as follows:

- **Is Generic Template:** This flag to be set to 'Y' if this workflow step will call GenericPaymentsService.
- **Has Extended Response:** Flag to set if extended response handler class needs to process the data for this workflow step.
- **Alert Required:** Flag to set if alert message needs to be generated.
- **Valid ID Request:** Comma separated valid idrequest(s) that can be source to this idrequest. Null: indicates any idrequest as source.
- **Audit Resource XSL:** Name of the user interface XSL file to generate the desired UI content for audit view, used in special cases where audit data is not HTML.
- **Audit Content Style:** Style ID to be used to generate the audit data content, used in special cases where audit data is not HTML. Each style id is mapped to respective content generator java class in <identity>.xml available in home folder of FCDB base directory.

Click **Next** to go to next step.

3. FCDB Service Registration

Business Services in FCDB consist of Java Interfaces and Classes implementing these interfaces. In FCDB, classes implementing these interfaces are also known as Business-End points. User needs to declare their business methods in these interfaces and implement the business logic in the corresponding endpoint class implementing these interfaces.

Data is transported to/from these Business Methods using Data Transfer Objects (DTO's). Each method in the Business Interface must have a Request DTO as an argument and Response DTO as return type.

Once the user has defined the Java Components they need to register it in table **MstServices**. Application will only load those services which are registered in this table.

MstServices Definition

Column Name	Type	Nullable	Default	Comments
NAMSERVICE	VARCHAR2(255 CHAR)	No		Name of the Service.
INTERFACE	VARCHAR2(255 CHAR)	No		Full Qualified name of the Interface Class.
ENDPOINT	VARCHAR2(255 CHAR)	No		Full Qualified name of the Service EndPoint Class implementing the Interface Class.
VERSION	NUMBER	No	0	Service Version.
ISENABLED	VARCHAR2(1 CHAR)	No	'Y'	Flag to enable/disable the service.
ISMULTIPHASE	VARCHAR2(1 CHAR)	No	'N'	Flag to be enabled if this is a MultiPhase Transaction.
ISROLLBACKONLY	VARCHAR2(1 CHAR)	No	'N'	Flag to be enabled if Transaction is to be rolled back after successful execution of the Service.
CUSTOM_HELPER	VARCHAR2(255 CHAR)	Yes		Full Qualified name of the Service Helper class implementing the interface "com.iflex.fcat.services.ServiceHelper".
NAMMETHOD	VARCHAR2(255 CHAR)	No		Method name defined in the Interface Class.
NAMTXNCONTEXT	VARCHAR2(255 CHAR)	Yes		
ISAPI	CHAR(1 CHAR)	No	'N'	Flag to be enabled if service can be exposed as an API.
ISAUDITREQUIRED	CHAR(1 CHAR)	Yes	'N'	Flag to enable Service Details Logging.
AUTHDTOMAPPER	VARCHAR2(255 CHAR)	Yes		Full Qualified name of the Authorization DTO Mapper class extending the class "com.iflex.fcat.services.apps.auth.authdtomapper.AbstractAuthDTOMapper".
TXNAUTHDTOMAPPER	VARCHAR	Yes		Full Qualified name of the Transaction Auth DTO

	2(255 CHAR)			Mapper class implementing the interface "com.iflex.fcat.services.hostinterface.HostDTOMapper".
VALIDATIONVERSION	NUMBER	No	0	Service Validation Version.
EXTENDEDRESPONSE HANDLER	VARCHAR 2(255 CHAR)	Yes		Full Qualified name of the Service Response Handler class extending the class "com.iflex.fcat.services.ExtResponseHandler".

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these FCDB Service Registrations.

To Open Service Configuration Wizard, Go to Menu

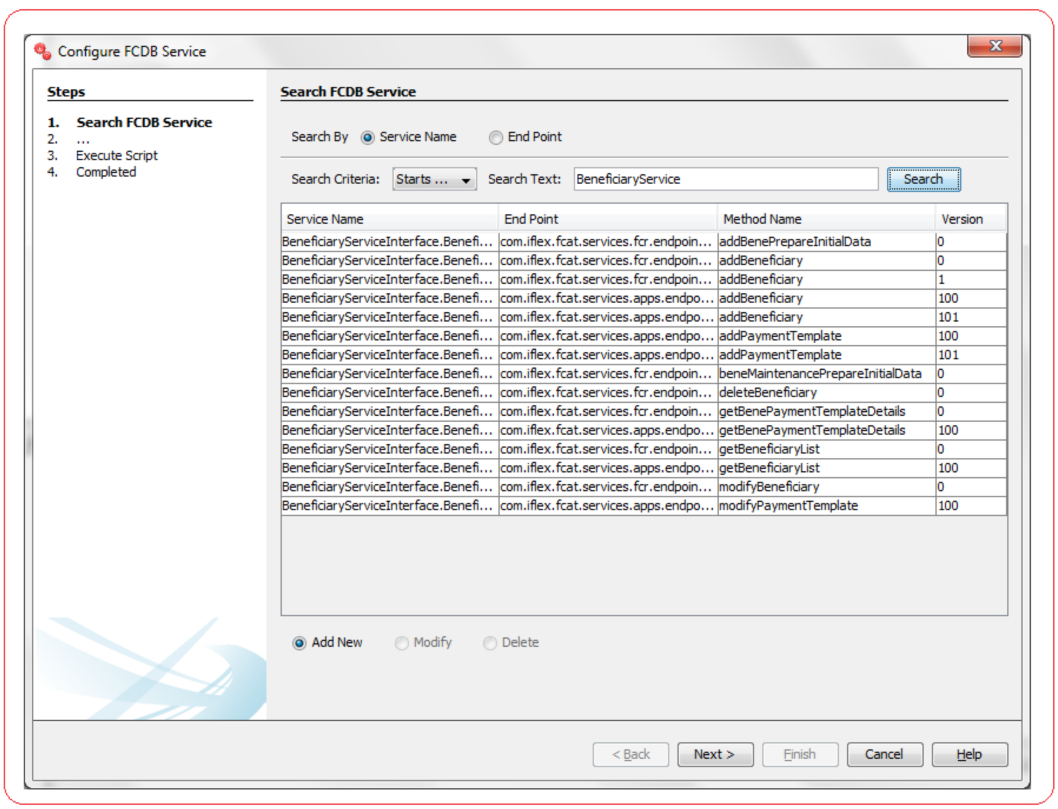
Tools → FCDB → Configure FCDB Service

Service Registration – Search

This step is meant to allow the user to search for the existing FCDB Services in **MstServices** table of the connected Database. One can use any of the following two parameters to filter the result:

- ✓ Service Name
- ✓ End Point

Select any of the above two parameters, select the search criteria, enter the search text and click on "Search" Button. A list of the available services as per the search criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria. You can add a new Service, Modify or Delete the existing Services.



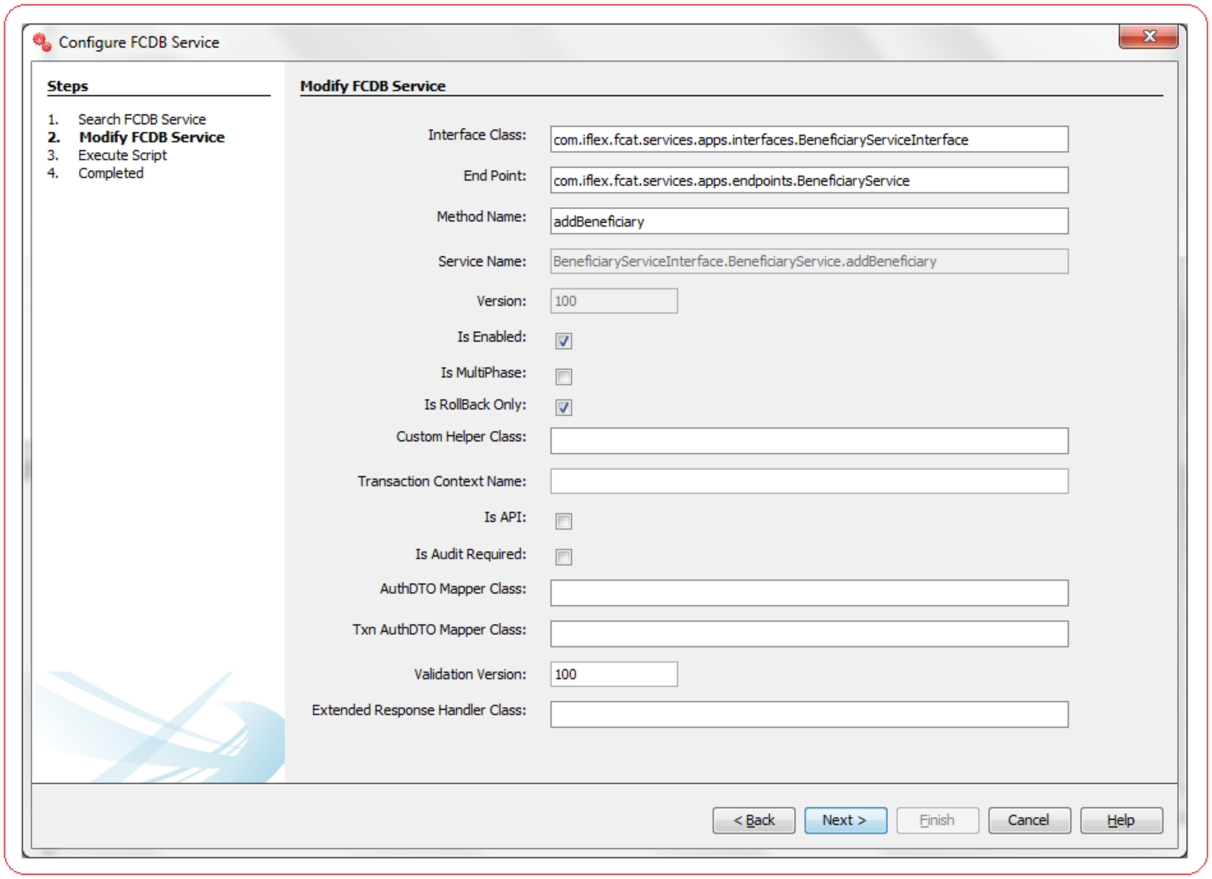
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add a new service in **MstServices** table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected service.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected service from the **MstServices** table.

Click **Next** to go to next step.

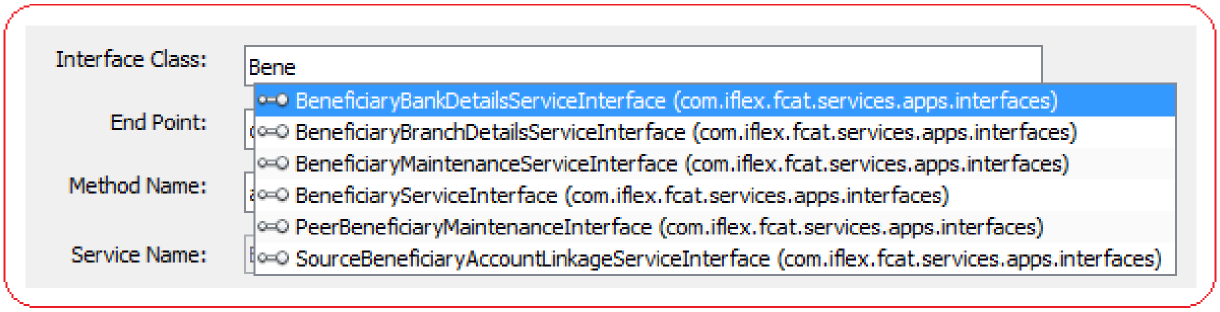
Service Registration – Edit

This step is meant to allow users to input details for a new service, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Interface Class:** Enter the Full Qualified name of the Service Interface Class. User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Interface names as shown below.



- **End Point:** Enter the endpoint to be used for this service. A Service Endpoint is a concrete implementation of a Service Interface. User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes implementing the entered Interface name.
- **Method Name:** Enter the name of the method to be invoked on executing this service. User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available methods defined in the interface.
- **Service Name:** Enter the service name for this service or use the auto-generated service name. The auto-generated service name is of the form <Interface>.<Endpoint>.<Method>.
- **Version:** Enter a valid numeric value as service version. Version is mandatory. Register the service with same service name but different version in order to extend a service.
- **Is Enabled:** Flag to indicate if this service is enabled or not.
- **Is Multiphase:** Flag to indicate a single-phase transaction context or a multi phase transaction.
- **Is RollBack Only:** Flag to be enabled if Transaction is to be rolled back after successful execution of the Service.
- **Custom Helper Class:** Enter a valid Full Qualified name of the Service Helper class implementing the interface **"com.iflex.fcat.services.ServiceHelper".** User can use auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available Classes implementing ServiceHelper Interface.
- **Is API:** Flag to be enabled if service can be exposed as an API.
- **Is Audit Required:** Flag to enable Service Details Logging.
- **AuthDTO Mapper Class:** Enter a valid Full Qualified name of the Authorization DTO Mapper class extending the class "**com.iflex.fcat.services.apps.auth.authdtomapper.AbstractAuthDTOMapper".** User can use auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available Classes extending AbstractAuthDTOMapper Class.
- **Txn AuthDTO Mapper Class:** Enter a valid Full Qualified name of the Transaction Auth DTO Mapper class implementing the interface "com.iflex.fcat.services.hostinterface.HostDTOMapper". User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes implementing HostDTOMapper Interface.
- **Validation Version:** Enter a valid numeric value as service validation version.
- **Extended Response Handler Class:** Enter a valid Full Qualified name of the Service Response Handler class extending the class "com.iflex.fcat.services.ExtResponseHandler". User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes extending ExtResponseHandler Class.

Click **Next** to go to next step.

4. ApplData

Oracle FLEXCUBE Direct Banking Application houses the table appldata where various application configuration are added which has a significant impact on the behavior of the application. A xml namedappldata.xml is also bundled with the application where all the records added in the database are replicated. The table in database and the xml bundled with the application always remain in sync so that no value configured by the user is lost.

Some Important Configurations done using ApplData

- ✓ **Internationalization:** Oracle FLEXCUBE Direct Banking Application can be adapted to different languages and technical requirements of a target Audience. Oracle FCDB Application is written in such a manner that it can be easily adapted to other languages and countries. This Internationalization is possible with the help of table appldata.
- ✓ **Application CSS Themes:** Oracle FLEXCUBE Direct Banking Application comes in different flavors of themes for the user to choose from. A theme is mapped to an entity and user type (Retail, Admin, etc). These mapping configurations are done in the table appldata. All the themes mapped to a particular entity and user type combination are available as a dropdown to user to choose from at the time of login. Oracle FLEXCUBE Direct Banking Application shall be launched in the theme selected by user.
- ✓ **Description and Summary:** Transactions, Services, Widgets, etc. are implemented in Oracle FLEXCUBE Direct Banking Application to achieve a defined task in application. The description and summary of these implemented components are written in the table appldata.
- ✓ **Grouping of Transactions:** A Transaction shown in the menu of Oracle FLEXCUBE Direct Banking Application is configured in the table appldata. Factors as to where to place a transaction in menu, its position in menu/submenu group, label of the transaction to be displayed, etc are all decided using the entries provided in appldata table.

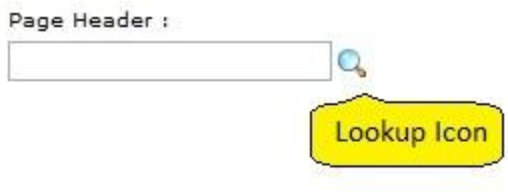
ApplData Definition

Column Name	Type	Nullable	Default	Comments
DATANAME	VARCHAR2(40 CHAR)	No		Data Name.
DATAVALUE	VARCHAR2(100 CHAR)	No		Data Value.
IDAPP	CHAR(8 CHAR)	No		App ID ('RR' or 'A1').
IDLANG	CHAR(3 CHAR)	No		Language ID.
IDDEVICE	CHAR(2 CHAR)	No		Channel/Device ID.
LOADFLAG	CHAR(1 CHAR)	Yes	'Y'	Flag to indicate if this appldata is to be loaded on startup or not.
VALUESTRING	VARCHAR2(1000 CHAR)	Yes		Value String.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these appldata values.

ApplData Wizard can be opened from the following options

1. **Using Menu Bar**: Go to Menu Options. Navigate through Tools → FCDB → Appl Data Wizard
2. **Using Lookup Icon**: Lookup icon is available against the fields in the Screen Layout Property panels. These lookup icons are available only against those fields where user is supposed to use XSL keyword as show below.



ApplData - Search

This step is meant to allow the user to search for the existing appldata configurations available in appldata table if connected to the Database, or retrieves values from the appldata.xml bundled with the tool.
A two criteria search is provided to user to search for existing ApplData values. User can use any of the following parameters to search for existing entries.

- ✓ **Data Name**
  Data Name column of appldata is treated as constants in Oracle FLEXCUBE Direct Banking application to differentiate between various ApplData records. For e.g. Data Name column takes the value TXN_DESC for an ApplData entry which provides description of a Transaction available in the application.

- ✓ **Data Value or Value String:** User can select one of the two parameters.

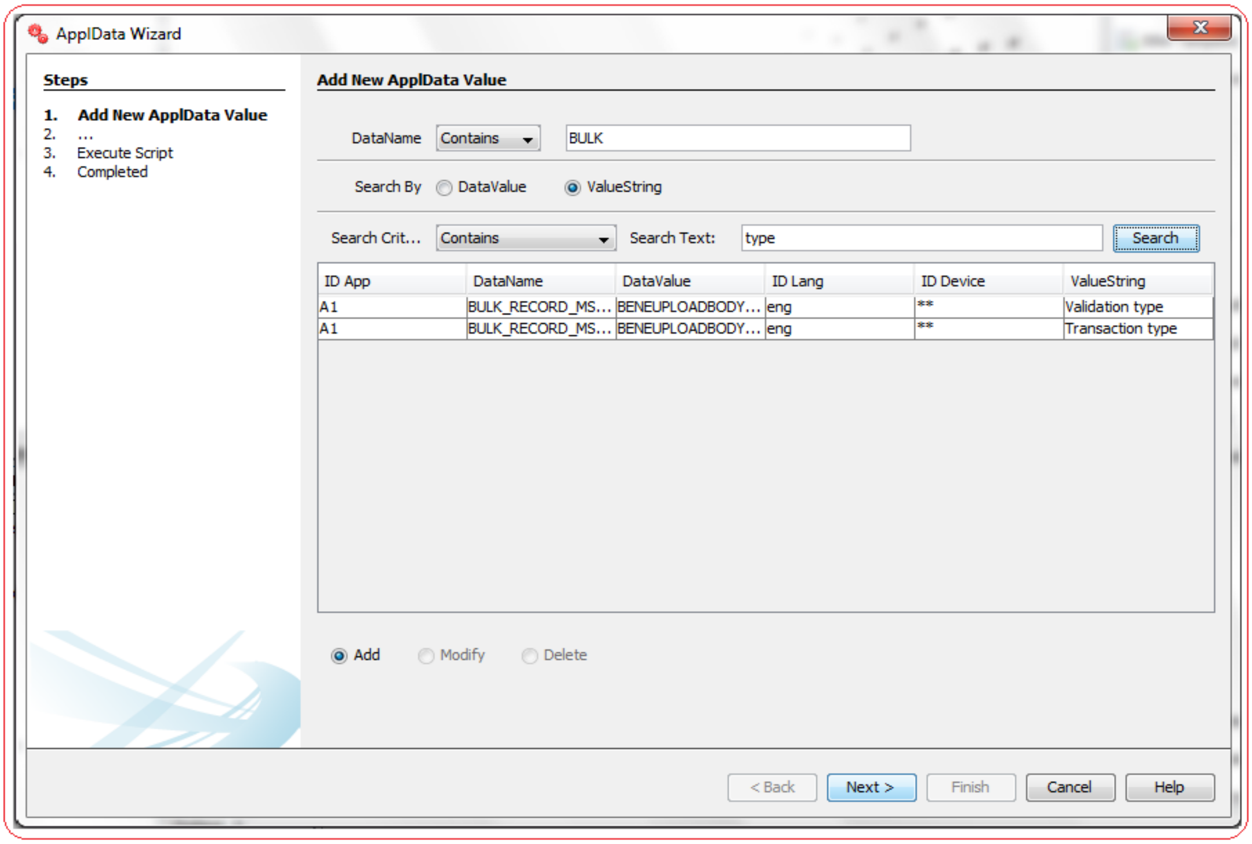
  - **Data Value:**
    The field Data Value is used for different purposes according to the value entered in field Data Name of table ApplData. For e.g. Data Value takes the value of Transaction ID for an ApplData entry that is configured to display the label of the Transaction in Oracle FLEXCUBE Direct Banking.

  - **Value String:**
    Value String contains the language dependent String that will used whenever its ApplData record is accessed in Oracle FLEXCUBE Direct Banking Application.

The search for the available appldata values is done using DataName and (DataValue or ValueString) as a whole. Enter the data value, select any of DataValue or ValueString parameter, select the search criteria from the drop down, enter the search text and click on "Search" Button.

It is always advisable to search for existing entries before adding a new appldata value in this table.

A list of the available configurations as per the search criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



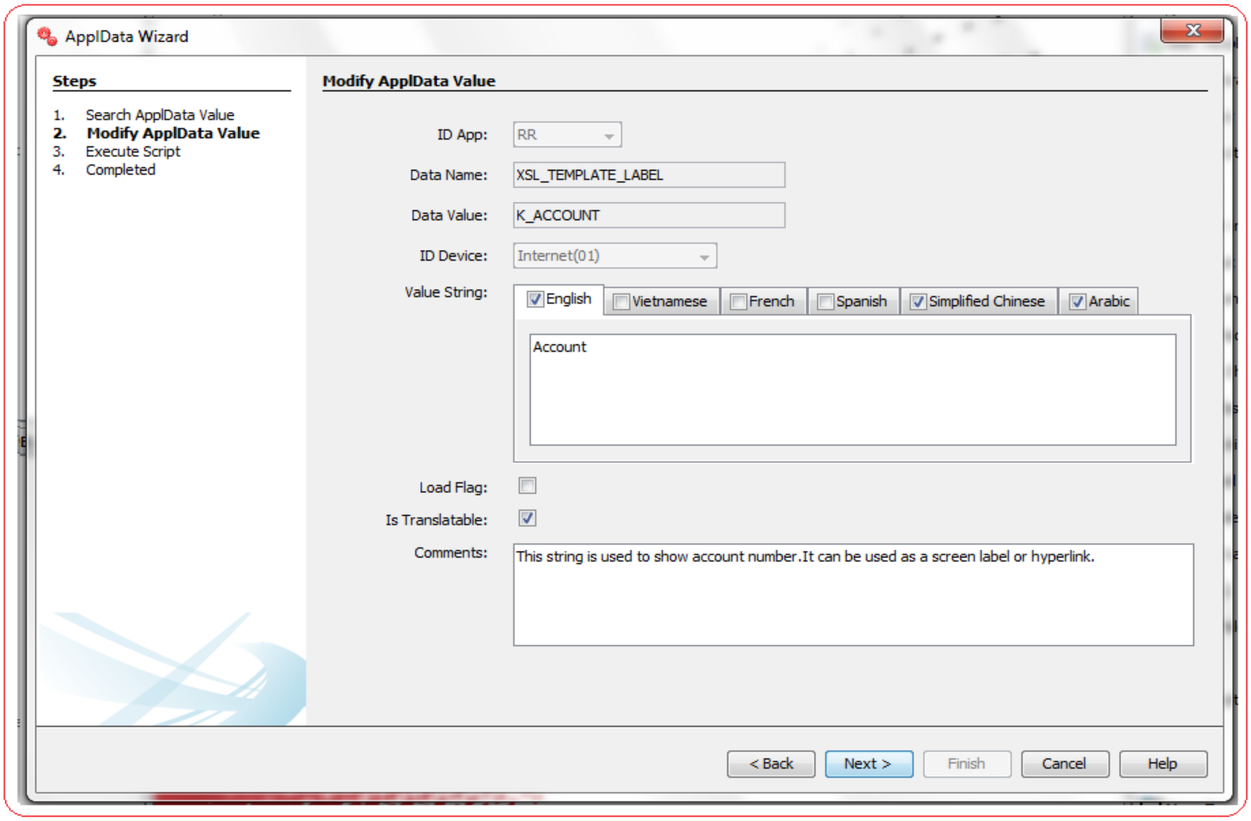
Depending upon the search output one can do the following operations

I. **Add New:** Select this radio button to add a new appldata value in appldata table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected appldata.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected appldata from the appldata table.

Click **Next** to go to next step.

ApplData - Edit

This step is meant to allow users to input details for a new appldata value for multiple languages, modify details of an existing entries or view details of an existing entries if user wants to delete the same. One can add/modify/delete existing ApplData entries for different languages in one go by checking the check box besides the needed languages.



List of the User Input fields and their description are as follows:

- **ID App:** Select the value from the drop down as ID App.
- **DataName**: Enter the valid String as Data Name. In case of screen design, data name should be XSL_TEMPLATE_LABEL.
- **Data Value**: Enter the valid String as Data Value. In case of screen design and data name is XSL_TEMPLATE_LABEL, then data value should start with K_.
- **ID Lang:** Select one or multiple languages for which you want this value to be displayed.
- **ID Device**: Select the channel for which you want this value to be displayed.
- **ValueString**: Enter a valid String as ValueString.
- **Load Flag**: Flag to indicate if this appldata is to be loaded on startup or not.
- **Is Translatable**: Flag to indicate if translation to be done.
- **Comments**: Enter a valid comment, if any for this message.

Click **Next** to go to next step.

5. Application Message

Oracle FLEXCUBE Direct Banking Application houses the table **applicationmessage** where various error, information, warning, success, debug messages are added which has a significant impact on the behavior of the application.

Some Important Configurations done using Application Message:

- ✓ **Internationalization:** Oracle FLEXCUBE Direct Banking Application can be adapted to different languages and technical requirements of a target Audience. Oracle FCDB Application is written in such a manner that it can be easily adapted to other languages and countries. This Internationalization of all the messages (error, warning, success etc.) is possible with the help of table **applicationmessage**.
- ✓ **Description and Summary:** Various error, information, warning, success, debug messages are used in Oracle FLEXCUBE Direct Banking Application to display the output to the user. The description and summary of these messages are written in the table **applicationmessage**.

ApplicationMessage Definition

Column Name	Type	Nullable	Default	Comments
IDAPP	CHAR(2 CHAR)	No		App ID ('RR' or 'A1').
IDMESSAGE	VARCHAR2(20 CHAR)	No		Message Identifier.
IDLANG	CHAR(3 CHAR)	No		Language ID.
IDDEVICE	CHAR(2 CHAR)	No		Channel/Device ID.
TXTMESSAGE	VARCHAR2(4000 CHAR)	No		Message text to be displayed in FCDB for this Message ID.
ISGUIENABLED	CHAR(1 CHAR)	No	'N'	GUI Enabled Flag.
MODULEID	VARCHAR2(20 CHAR)	Yes		Module ID of FCDB if any.
ISTRANSLATABLE	CHAR(1 CHAR)	No	'Y'	Translation Flag.
COMMENTS	VARCHAR2(4000 CHAR)	Yes		Comments if any.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these application messages.

To Open Application Message Wizard, Go to Menu Options

Tools → FCDB → Application Message Wizard

Application Messages – Search

This step is meant to allow the user to search for the existing application messages configured in **'applicationmessage'** table if connected to the Database. A two criteria search is provided to user to search for existing **applicationmessage** values.

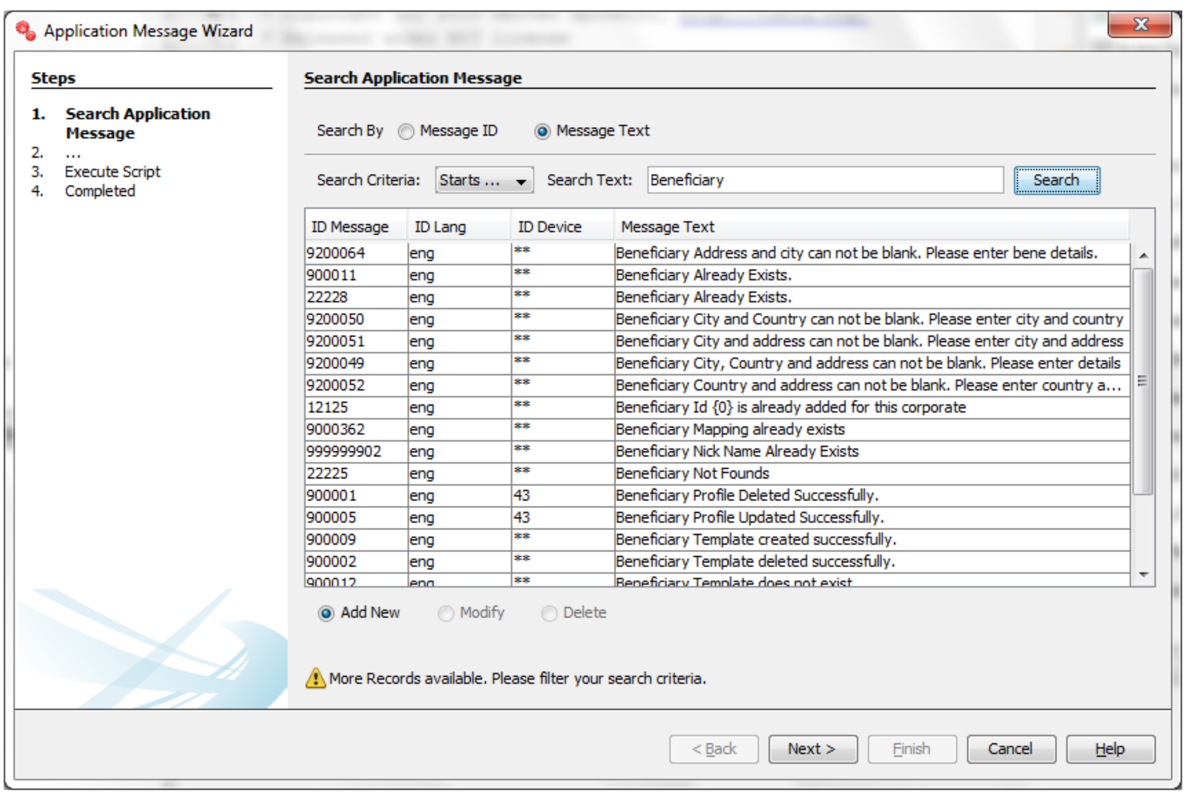
User can use any of the following parameters to search for existing entries.

- ✓ **Message Id:** Message Id is unique identifier used to identify the text message.
- ✓ **Message Text:** Message text that will be used in Oracle FLEXCUBE Direct Banking Application to display the error, warning, success messages.

The search for the available **'applicationmessage'** values is done using **Message Id** and **Message Text**. Select any of Message Id or Message Text parameter, select the search criteria from the drop down, enter the search text and click on "Search" Button.

It is always advisable to search for existing entries before adding a new applicationmessage value in this table.

The list of available messages as per the search criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



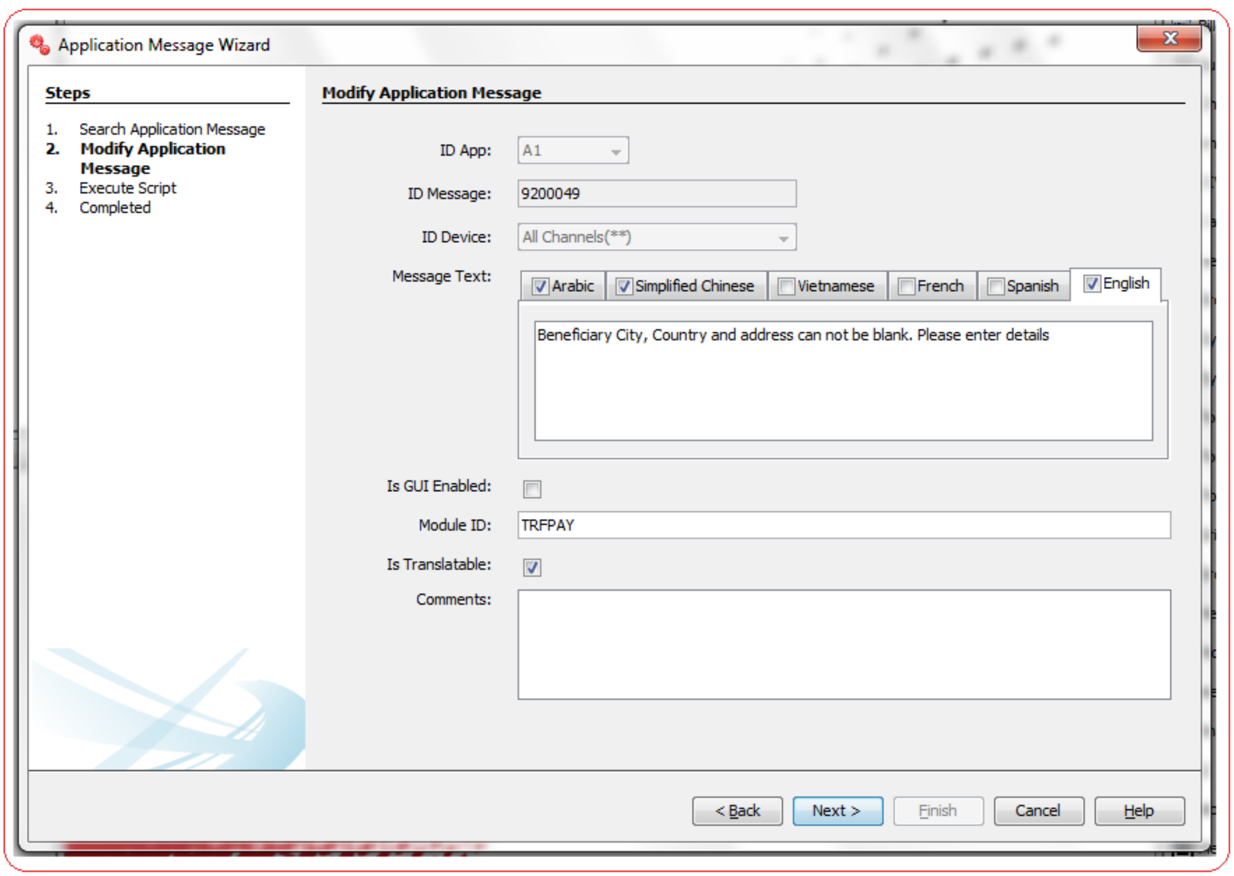
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new application message.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Application Messages – Edit

This step is meant to allow users to input details for a new application message, modify details of an existing one or view details of an existing one if user wants to delete the same. One can add/modify/delete existing Application Message entries for different languages in one go by checking the check box besides the needed languages.



List of the User Input fields and their description are as follows:

- **ID App:** Select the value from the drop down.
- **ID Message:** Give a valid numeric value as Message ID.

- **ID Lang:** Select one or multiple languages for which you want this message to be displayed.
- **ID Device:** Select the channel for which you want this message to be displayed.
- **Message Text:** Enter the message text to be displayed.
- **Is GUI Enabled:** Flag to make this property GUI enabled. This will allow FCDB Admin users to edit the property from FCDB application itself.
- **Module ID:** Specify the module id, if any for which this message is being used.
- **Is Translatable:** Flag to indicate if translation to be done.
- **Comments:** Enter a valid comment, if any for this message.

Click **Next** to go to next step.

6. MstHostInterface

Oracle FLEXCUBE Direct Banking Application can talk to external Host System using Host Interface Framework. All HostInterface invocations need to be registered in table **msthostinterface**. This table holds the relevant information for invoking a host system.

MstHostInterface Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2(5 CHAR)	No		Entity id.
IDHOST	VARCHAR2(5 CHAR)	No		Host Id.
IDREQUEST	VARCHAR2(50 CHAR)	No		Request Id.
VERSION	NUMBER	No		Version Number.
REQUEST_IFACE_MAPPER	VARCHAR2(255 CHAR)	Yes		Full Qualified name of the Mapper Class implementing the Interface "com.iflex.fcat.services.hostinterface.HostDTOMapper".
RESPONSE_IFACE_MAPPER	VARCHAR2(255 CHAR)	Yes		Full Qualified name of the Mapper Class implementing the Interface "com.iflex.fcat.services.hostinterface.HostDTOMapper".
INTERFACE_IMPL	VARCHAR2(4000 CHAR)	No		Full Qualified name of the HostInterfacing Class implementing the interface "com.iflex.fcat.services.hostinterface.HostAdapter".
ISENABLED	CHAR(1 CHAR)	No	'Y'	Flag to enable/disable this interface.
ADTNL_PARAMS	VARCHAR2(4000 CHAR)	Yes		Text for setting Additional Parameters like Interfacing Mode(JMS/EJB) , request-response XSL's e.t.c.
IDINTERFACE	VARCHAR2(20 CHAR)	No		A unique identifier for this Interface Entry.
REF_ID_ENTITY	VARCHAR2(300 CHAR)	Yes		Comma separated list of entities to which the same interface is to be copied.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these application messages.

To Open Application Message Wizard, Go to Menu Options

Tools → FCDB → MstHostInterface Wizard

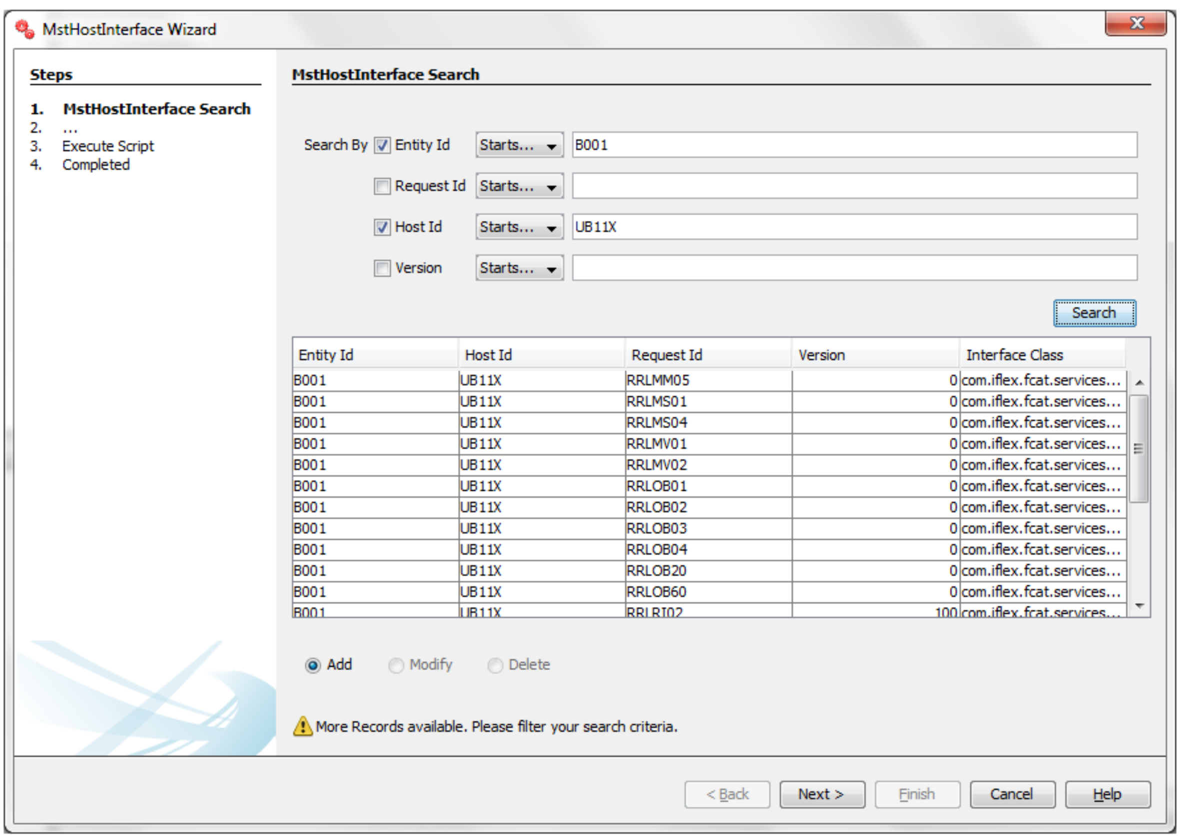
MstHostInterface – Search

This step is meant to allow the user to search for the existing entries available in msthostinterface table of the connected Database. It is always advisable to search for existing entries before adding a new entry in this table.

One can use any of the following four parameters to filter the result:

- ✓ Entity id
- ✓ Request Id
- ✓ Host Id
- ✓ Version

Select any of the above parameters, enter the search text and click on "Search" Button. The list of available results as per the search criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



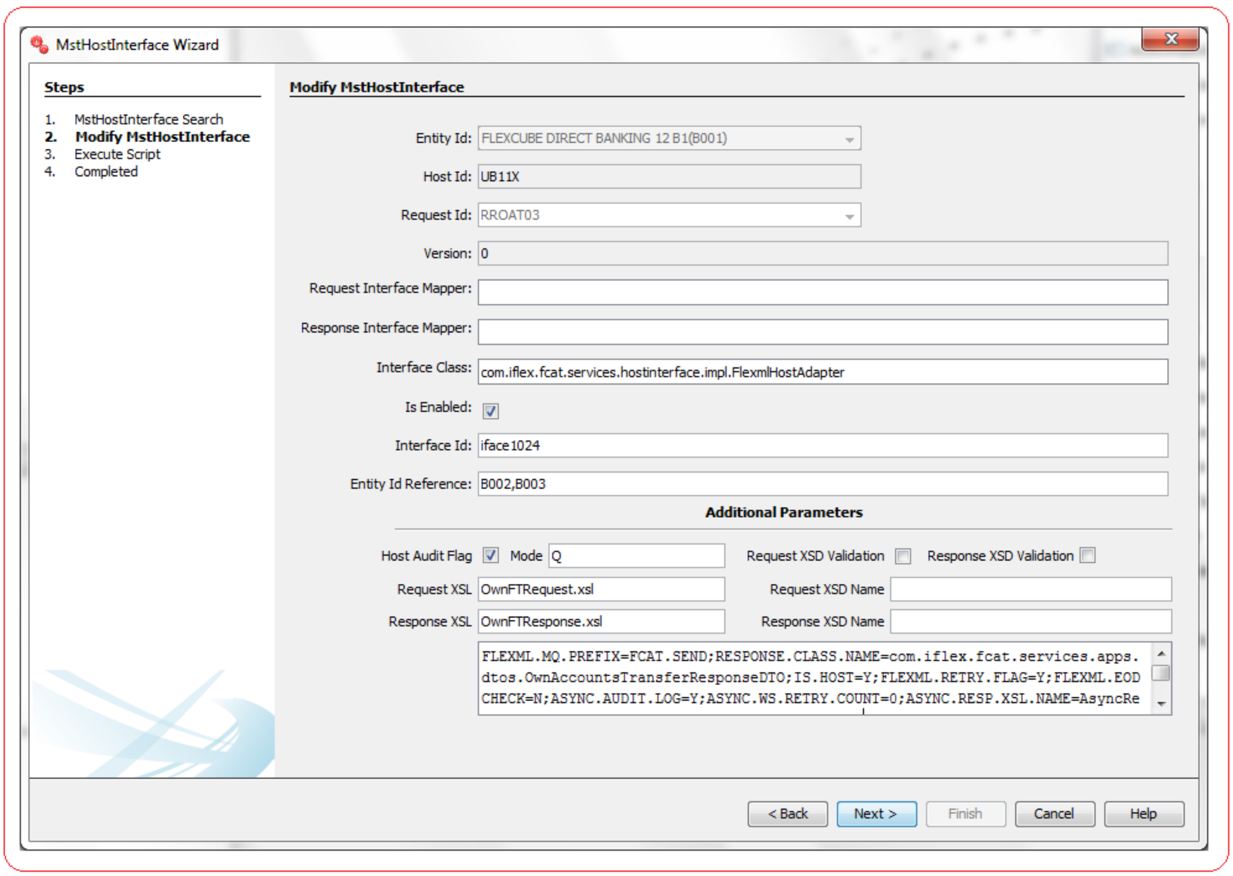
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new Host interface.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstHostInterface – Edit

This step is meant to allow users to input details for a new msthostinterface entry, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Entity Id:** Enter the Entity id for which the configuration is to be maintained.
- **Host Id:** Enter the Host Id which is to be invoked.

- **Request Id:** Enter the Request Id for which the interface is to be invoked.
- **Version:** Enter the Version of the Interface.
- **Request Interface Mapper:** Enter the Full Qualified name of the Mapper Class implementing the Interface "com.iflex.fcat.services.hostinterface.HostDTOMapper". This class is to convert the RequestDTO available from Service tier into a Request DTO understandable by adapter. User can use auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available Mappers.
- **Response Interface Mapper:** Enter the Full Qualified name of the Mapper Class implementing the Interface "com.iflex.fcat.services.hostinterface.HostDTOMapper". This class is to convert the ResponseDTO available from adapter into a Response DTO understandable by Service Tier. User can use auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available Mappers.
- **Interface Class:** Enter the Full Qualified name of the HostInterfacing Class implementing the interface "com.iflex.fcat.services.hostinterface.HostAdapter". User can use auto-suggest on this field by pressing **Ctrl+Space** and selecting from the list of available HostInterfacing Classes.
- **Is Enabled:** Flag to enable/disable this interface.
- **Additional Parameters:** Enter the text for setting Additional Parameters like Interfacing Mode (JMS/EJB), request-response XSL's e.t.c. For more details on these additional parameters please refer to **"Oracle_FLEXCUBE_Direct_Banking_System_Handbook_Volume_V.pdf".**
- **Interface Id:** A unique identifier for this Interface Entry.
- **Entity Id Reference:** Comma separated list of entities to which the same interface is to be copied.

Click **Next** to go to next step.

7. MstProperties

Oracle FLEXCUBE Direct Banking Application houses all application properties required for configurations in the database table **mstproperties**.

Service tier in Oracle FCDB uses properties file **fcat.properties** and table **mstproperties** to read application properties. **"fcat.properties"** file typically contains only database information and all the other application configurations are available in table **mstproperties**.

MstProperties Definition

Column Name	Type	Nullable	Default	Comments
IDSERVER	VARCHAR2(59 CHAR)	No		Server Identifier.
PROPNAME	VARCHAR2(255 CHAR)	No		Property Name.
PROPVALUE	VARCHAR2(4000 CHAR)	Yes		Property Value.
ENABLED	CHAR(1 CHAR)	Yes	'N'	Enabled Flag.
ISMODIFIED	CHAR(1 CHAR)	Yes	'Y'	Modified Flag
ISGUIENABLED	CHAR(1 CHAR)	Yes	'N'	GUI Enabled Flag.
DATEFFECTIVE	DATE	Yes	System Date	Effective Date.
DATMODIFIED	DATE	Yes	System Date	Modified Date.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open MstProperties Registration Wizard, Go to Menu Options

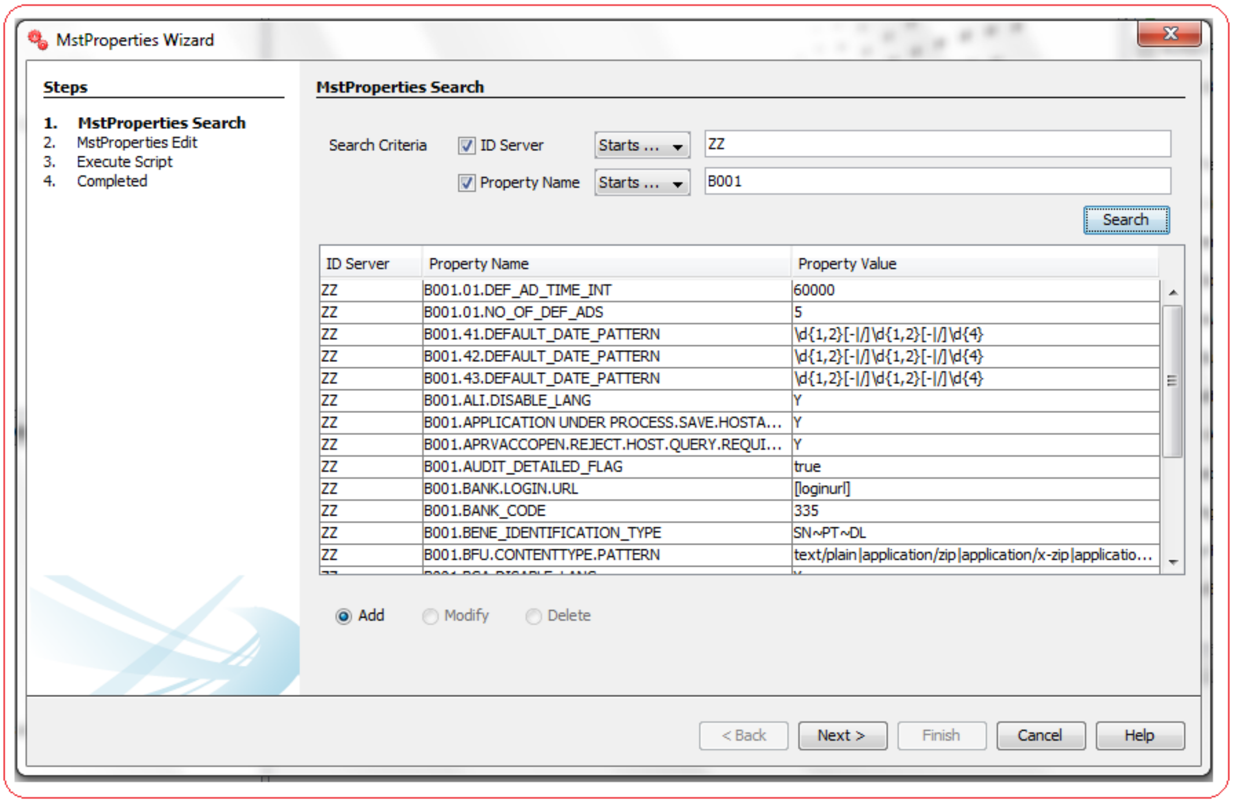
Tools → FCDB → MstProperties Registration

MstProperties - Search

This step is meant to allow the user to search for the existing entries available in mstproperties table of the connected Database. It is always advisable to search for existing entries before adding a new property in this table. One can use either or both the parameters to filter the result described below:

- ✓ ID Server
- ✓ Property Name

Check the parameters to search mstproperties with, select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available properties in the table **mstproperties** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



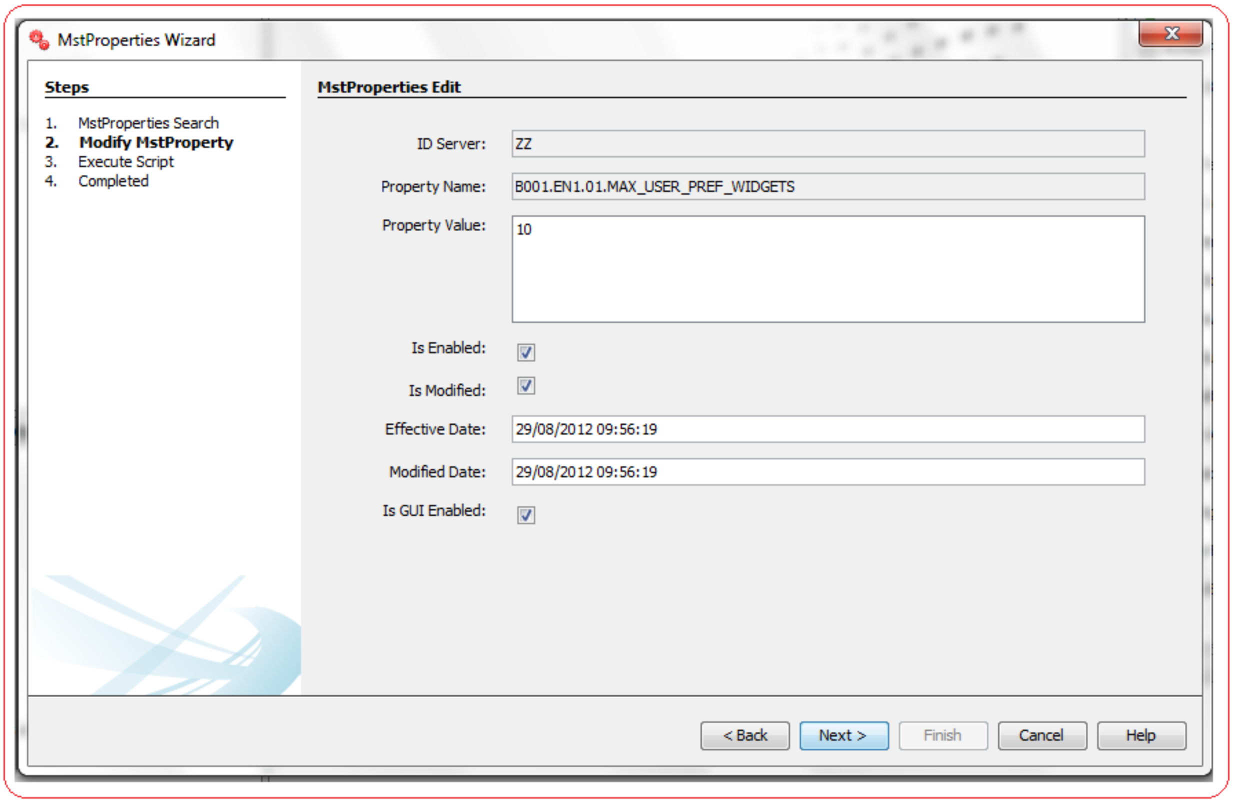
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new property in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstProperties – Edit

This step is meant to allow users to input details for a new mstproperty, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **ID Server:** Enter the server identifier to be used. Default id 'ZZ' should be used.
- **Property Name:** Enter the name of this property. E.g. Property Name to allow the Maximum number of Widgets for a user is MAX_USER_PREF_WIDGETS. Entity/User Type/Channel can be specialized for particular property using property name as <ENTITY_ID>.<USERTYPE>.<IDCHANNEL>.MAX_USER_PREF_WIDGETS'
- **Property Value:** Enter the value for this property. E.g. it may be some numeric value to allow the Maximum widgets a user can select.
- **Enabled:** Flag to enable and make this property available.
- **Is Modified:** Flag to indicate if this property is modified or not.
- **Effective Date:** Effective date when this property was added.
- **Modified Date:** Modification date when this property was modified.
- **Is GUI Enabled:** Flag to make this property GUI enabled. This will allow FCDB Admin users to edit the property from FCDB application itself.

Click **Next** to go to next step.

8. Data Dictionary Registration

Data Dictionary Types

The term Data Types in Oracle Flexcube Direct Banking identifies the types of the various data field elements in a request. The standard data types supported by the Oracle FCDB system are defined in the table **data_dictionary_types.**

FCDB supports a list of standard data types to be used and have the flexibility to support custom data types. All the standard data types and custom data types are stored in the **data_dictionary_types** table.

For Example, The Account Number can be treated as a new data type with specific validator associated with the data type to only validate for the rules for account numbers. Similarly, Credit Card can be a new data type defined.

Data Dictionary Types Definition

Column Name	Type	Nullable	Default	Comments
DATATYPE	CHAR(20 CHAR)	No		Data Type. Primary Key of table.
DATATYPE_DESC	VARCHAR2(1020 CHAR)	Yes		Description of Data Type.
DATATYPE_VALIDATOR	VARCHAR2(1020 CHAR)	Yes		Validator to be used for Data Type.
POOLSIZE	NUMBER(22)	Yes	3	Pool Size.
BASE_TYPE	VARCHAR2(40 CHAR)	Yes		Base Type of Data Type ID.
COMMENTS	VARCHAR2(1020 CHAR)	Yes		Comments if any.

Data Dictionary

The Data Dictionary is the master set of data elements used within the application. In Oracle Flexcube Direct Banking, these Data Dictionary values are maintained in the table data_dictionary.

The Data Dictionary identifies the data type of the base data element and also the attributes like the minimum and maximum length of the data. Data Dictionary elements can be reused for multiple fields hence applying common validations to fields across various requests.

Data Dictionary Definition

Column Name	Type	Nullable	Default	Comments
FIELDNAME	VARCHAR2(1020 CHAR)	No		Field Name of Data Dictionary. Primary Key of table.
FIELDDESC	VARCHAR2(400 CHAR)	Yes		Field Description of Data Dictionary.
FIELDTYPE	CHAR(12 CHAR)	Yes		Field Type of Data Dictionary. Foreign Key reference Data Type of table data_dictionary_types
MINLENGTH	NUMBER(22)	Yes		Minimum Length.
MAXLENGTH	NUMBER(22)	Yes		Maximum Length.

FIELDFORMAT	VARCHAR2(1020 CHAR)	Yes		Field Format.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these Transactions in the above tables.

To Open Data Dictionary Registration Wizard, Go to Menu Options

Tools → FCDB → Validation Engine → Data Dictionary Registration

Data Dictionary - Search

This step is meant to allow the user to search for the existing Data Types supported by Oracle FCDB in table **data_dictionary_types** or existing Data Dictionary values in table **data_dictionary** of the connected Database.

A two criteria search is provided to user to search for existing values. User can use any of the following parameters to search for existing entries. Select either of the tables.

- ✓ Data Dictionary Types: The Following Parameters are provided to user to search for existing Data Types in **'data_dictionary_types'** table.
  a) **Data Type:** Unique ID of Data Type.
  b) **Description:** Description of Data Type.
- ✓ Data Dictionary: The Following Parameters are provided to user to search for existing Data Dictionary values in **'data_dictionary'** table.
  a) **Data Type:** Field Type of Data Dictionary Value.
  b) **Field Name:** Unique Field Name of Data Dictionary Value.
  c) **Description:** Description of Data Dictionary Value.

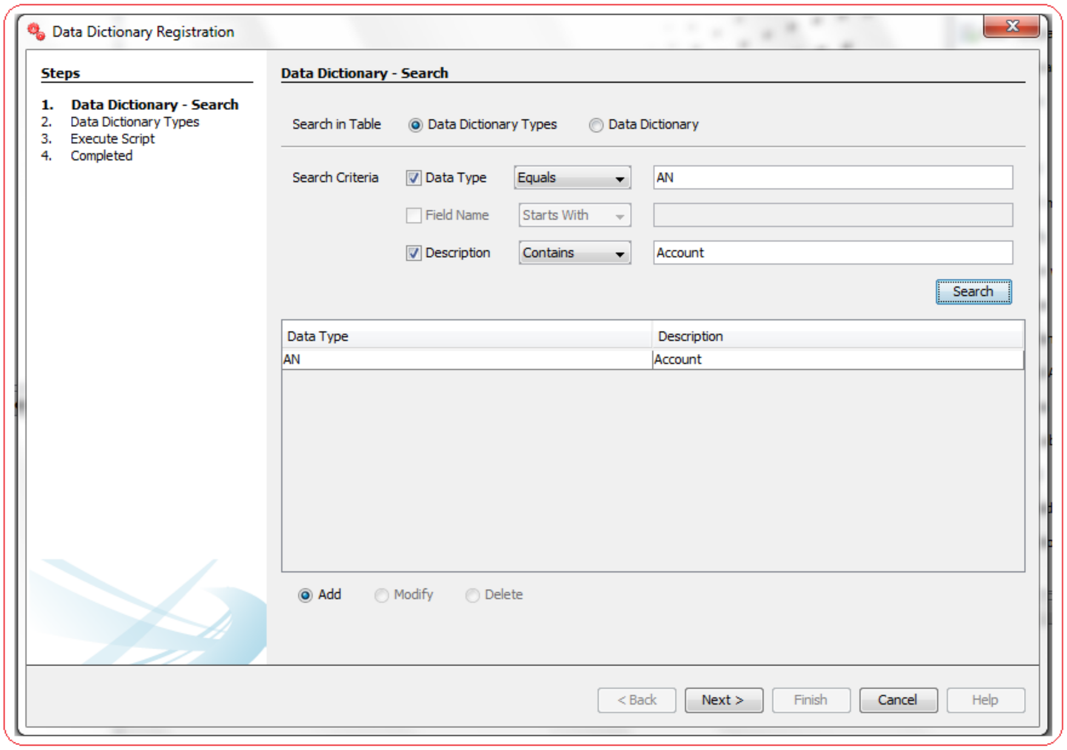
Tick the parameters to search with, select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available Data Dictionary Types or Data Dictionary values in the corresponding selected table matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.

Depending upon the search output one can do the following operations,

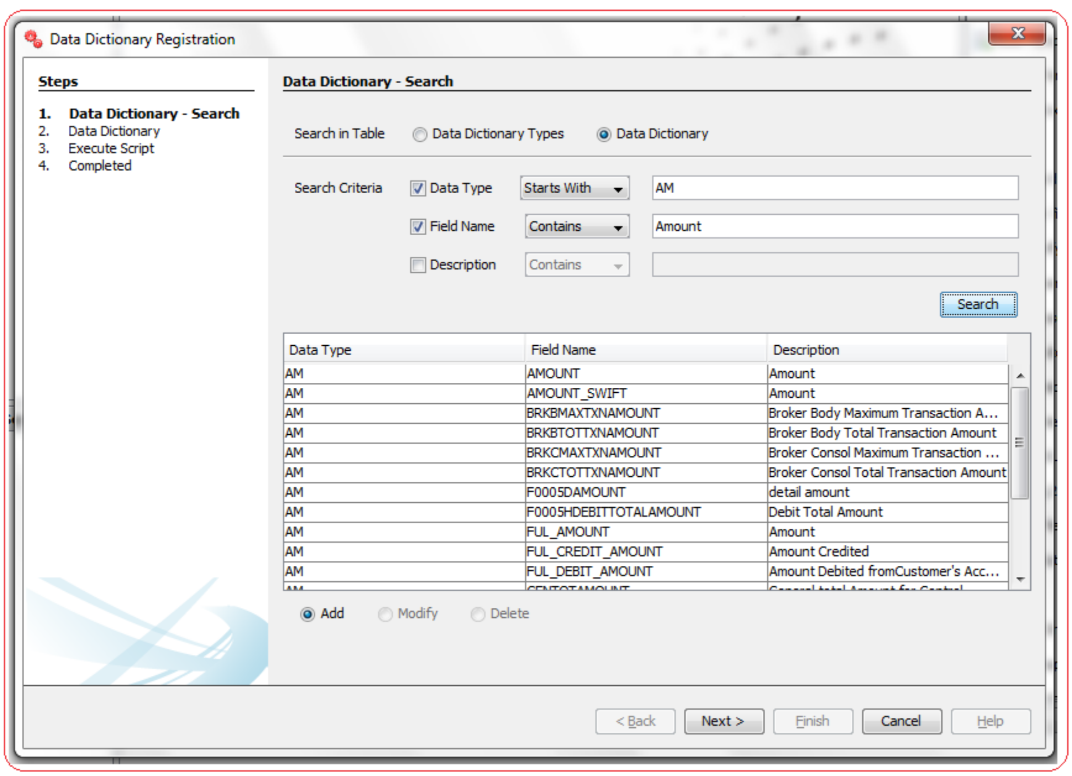
I. **Add New:** Select this radio button to add new Data Dictionary Type or Data Dictionary value in the selected table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Data Dictionary Types – Search



Data Dictionary – Search

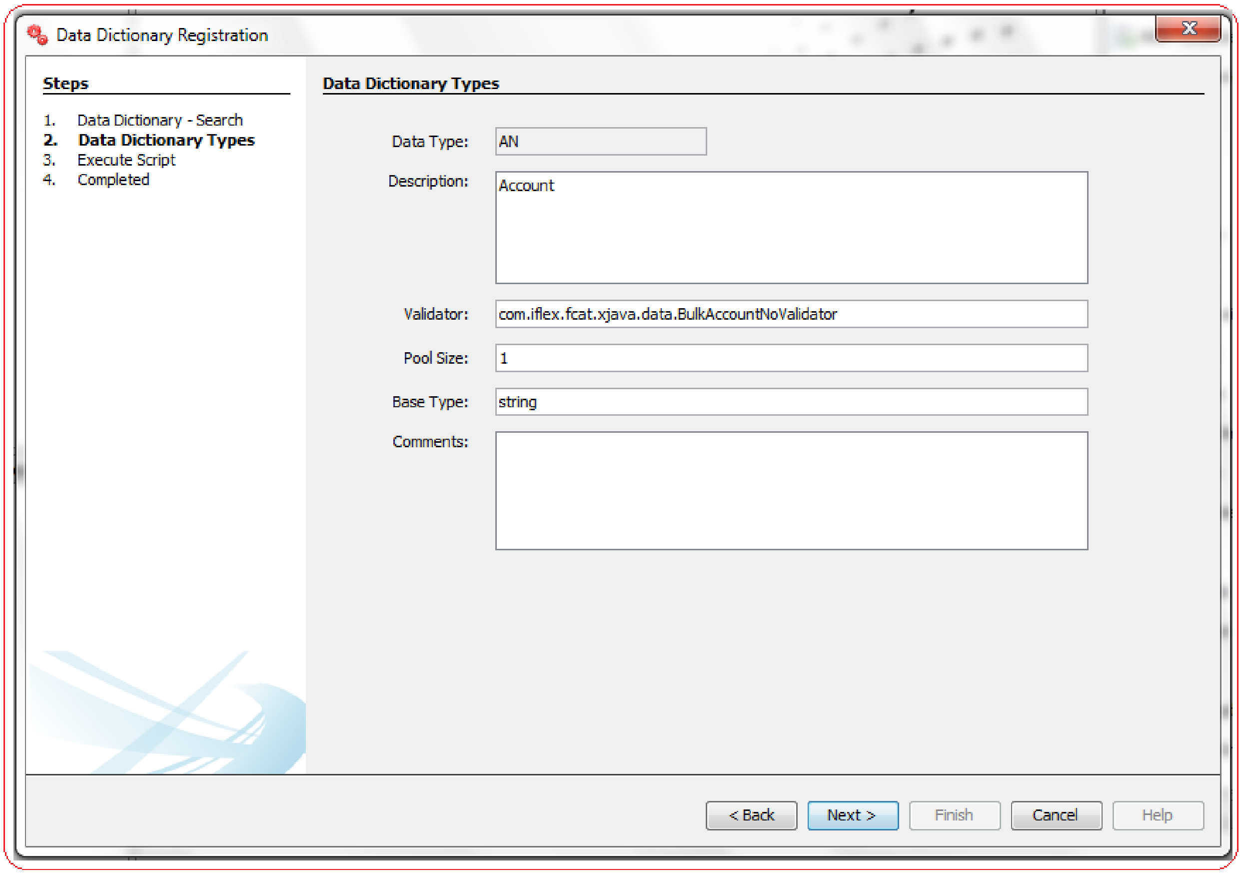


Data Dictionary – Edit

This step is meant to allow users to input details for a new data type or data dictionary record in **data_dictionary_types / data_dictionary** table, modify details of an existing one or view details of an existing one if user wants to delete the same.

Data Dictionary Types

A panel showing/asking for the details will be displayed. Enter the Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the data_dictionary_types table.



List of the User Input fields and their description are as follows:

- **Data Type:** Enter a ID that will uniquely identify this Data Type. It is Mandatory.
- **Description:** Enter the description of the Data Type.
- **Validator:** Enter a Validator for this Data Type. Validator provides the required validation logic for the data type and data field validations.
- **Pool Size:** Enter a valid Integer as Pool Size. Default Value is 3.
- **Base Type:** Enter the Base Data Type for this Data Type.
- **Comments:** Enter Comments if any for future reference.

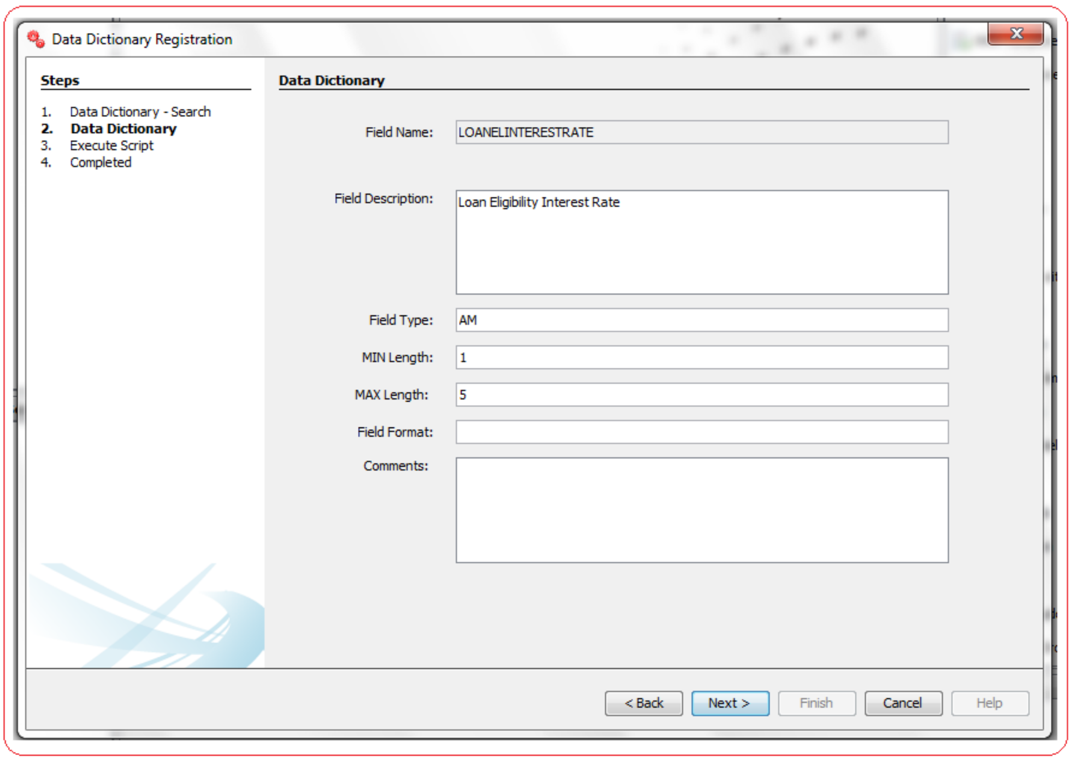
Some of the data types currently supported in Oracle FLEXCUBE Direct Banking are:

- **ALPHABETS (A):** This data type allows values which contain only Alphabets in upper case and lower case. Special characters and Numeric digits are not allowed for this data type.
- **STRING (S):** This data type allows values which contain Alphabets and Numeric Digits. Special characters are not allowed, by default, for this data type.
- **NUMERIC (N):** This data type allows all numeric values for a given data field. This includes all integer and floating point values.
- **FLOATING POINT (F):** This data type allows all numeric values for a given data field. This includes all integer and floating point values.
- **EMAIL (E):** This data types is used to validate for data fields used for email addresses. The validator checks for the required valid fields (e.g.: @ and .) at appropriate locations and allows only characters allowed as per the SMTP specification in the email address.
- **DATE (D):** The date type supports validations for date fields. The default incoming date format is expected to be DD/MM/YYYY.
- **TIMESTAMP (T):** This data type is exactly similar to the DATE data type but differs only in the instance of the typed object returned by the validator. The validator for the TIMESTAMP data type returns the actual value as a java.sql.Timestamp instance while the DATE validator returns an instance of java.util.Date.

Click **Next** to go to next step.

Data Dictionary

A panel showing/asking for the details will be displayed. Enter the Details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the data_dictionary Table.



List of the User Input fields and their description are as follows:

- **Field Name:** Enter an ID that will uniquely identify this Data Dictionary value. It is Mandatory.
- **Field Description:** Enter Description of Data Dictionary Value.
- **Field Type:** Enter the Field Type for Data Dictionary value. This field is the foreign key referencing Data Type field of Data Dictionary Type Table.
- **MIN Length:** Enter a valid integer as Minimum Length of Data Dictionary value.
- **MAX Length:** Enter a valid integer as Maximum Length of Data Dictionary value.
- **Field Format:** Enter a valid string as Field Format. The interpretation of the FIELDFORMAT field in the definition is as per the data type and the validator for the data types interprets that field for completing the validations. For Example, If the this field for a Reference Number data field indicated as a POSITIVE NUMBER contains the values 9999, the value greater than 9999 would result in a validation error.
- **Comments:** Enter Comments if any for future reference.

Click **Next** to go to next step.

9. Txn Data Registration

Validation Templates are defined and maintained in the **txn_data_master** and **txn_data** database tables. The Validation template identifies the request data fields and tags them to the Data Dictionary and the Data Types defined. The template also identifies whether a field is mandatory in the request or not and also whether a field requires data validation or not.

All Validation Templates in FCDB is maintained in the following 2 tables,

- ✓ **txn_data_master**
- ✓ **txn_data**

Txn Data Master Definition

Column Name	Type	Nullable	Default	Comments
IDREQUEST	VARCHAR2(255 CHAR)	No		Request ID. Primary Key of table.
TXN_DESC	VARCHAR2(255 CHAR)	Yes		Description of Transaction.
CUSTOM_VALIDATOR	VARCHAR2(1000 CHAR)	Yes		Custom Validator.
NAMSCHEMA	VARCHAR2(255 CHAR)	Yes		Schema Name.
TYPEREQUEST	CHAR(1 CHAR)	Yes		Request Type.
REFREQUEST	VARCHAR2(2000 CHAR)	Yes		Reference Request.
COMMENTS	VARCHAR2(4000 CHAR)	Yes		Comments.
ISFINREQUEST	CHAR(1 CHAR)	Yes		Fin Request Flag.
TOTALFIELDS	NUMBER	Yes		Total Fields.
MODEREQUEST	NUMBER	Yes	0	Request Mode.
SEPSTRING	VARCHAR2(10 CHAR)	Yes		Separator String.
ENCODING	VARCHAR2(20 CHAR)	Yes		Encoding.
PREFIX	VARCHAR2(15 CHAR)	Yes		Prefix.
POSTFIX	VARCHAR2(10 CHAR)	Yes		Postfix.
TERMINATOR	VARCHAR2(255 CHAR)	Yes		Terminator.
ADJUSTMENT	NUMBER	Yes		Adjustment.
BIZTYPE	CHAR(1 CHAR)	Yes		Business Type.
FREETEXT	VARCHAR2(2000 CHAR)	Yes		Free Text.
ISLENIENT	CHAR(1 CHAR)	Yes	'N'	Lenient Flag.

Txn Data Definition

Column Name	Type	Nullable	Default	Comments
IDREQUEST	VARCHAR2(255 CHAR)	No		Request ID. Composite Primary Key
FLD_NUMSEQUENCE	NUMBER	No		Number Sequence. Composite Primary Key
FIELDNAME	CHAR(12 CHAR)	Yes		Field Name.
REFFIELDNAME	VARCHAR2(50 CHAR)	Yes		Reference Field Name.
FIELDFORMAT	VARCHAR2(255 CHAR)	Yes		Field Format.

VAL_ENUM	VARCHAR2(255 CHAR)	Yes		Validation Enumeration.
DEFAULT_VALUE	VARCHAR2(4000 CHAR)	Yes		Default Value.
CUSTOM_VALIDATOR	VARCHAR2(255 CHAR)	Yes		Custom Validator.
ISREQUESTFLD	CHAR(1 CHAR)	Yes	'Y'	Request Field Flag.
VALIDATION_REQUIRED	CHAR(1 CHAR)	Yes	'Y'	Validation Required Flag.
ISMANDATORY	CHAR(1 CHAR)	Yes	'Y	Mandatory Flag.
ERRORCODE	VARCHAR2(20 CHAR)	Yes		Error Code.
TOKEN1	VARCHAR2(255 CHAR)	Yes		Token 1.
TOKEN2	VARCHAR2(255 CHAR)	Yes		Token 2.
TOKEN3	VARCHAR2(255 CHAR)	Yes		Token 3.
LENGTH	NUMBER	Yes		Length.
FIXEDLENFLAG	CHAR(1 CHAR)	Yes		Fixed Length Flag.
JUSTIFICATION	CHAR(2 CHAR)	Yes		Justification.
FILLCHAR	CHAR(1 CHAR)	Yes		Fill Char.
DELIMITER	VARCHAR2(30 CHAR)	Yes		Delimiter.
PREFIXLEN	NUMBER	Yes		Prefix Length
POSTFIXLEN	NUMBER	Yes		Postfix Length
TYPFIELD	CHAR(1 CHAR)	Yes		Field Type.
NUMOCC	NUMBER	Yes		Num OCC.
IDOCCFIELD	NUMBER	Yes		ID OCC Field.
ISPARAMFIELD	CHAR(1 CHAR)	Yes	'N'	Parameter Field Flag.
ISPLACEHOLDER	CHAR(1 CHAR)	Yes	'N'	Place Holder Flag.
ENRICHMENT	VARCHAR2(255 CHAR)	Yes		Enrichment.
ISAGGREGATE	CHAR(1 CHAR)	Yes	'N'	Aggregate Flag.
MULTIPLIER	NUMBER	Yes		Multiplier.
MINOCCUR	NUMBER	Yes		Minimum Occurrence.
MAXOCCUR	NUMBER	Yes		Maximum Occurrence.
ISHASHFIELD	CHAR(1 CHAR)	Yes	'N'	Hash Field Flag.
FREETEXT	VARCHAR2(2000 CHAR)	Yes		Free Text.
DIVIDER	NUMBER	Yes	-1	Divider.
ENPARAMFIELDS	VARCHAR2(2000 CHAR)	Yes		Enrichment Parameter Fields.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these Txn Data values in the above tables.

To Open Txn Data Registration Wizard, Go to Menu Options

Tools → FCDB → Validation Engine → Txn Data Registration

Txn Data Registration - Search

This step is meant to allow the user to search for the existing Txn Data Master and Txn Data values available respectively in tables given below of the connected Database.

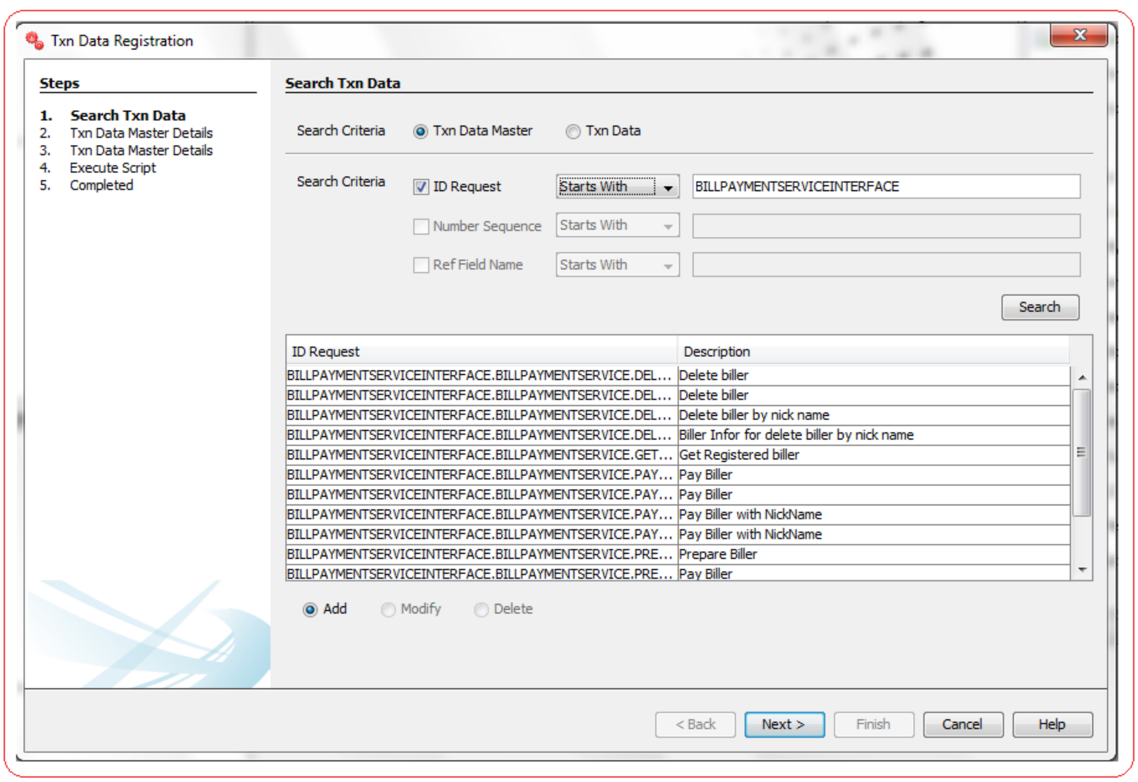
- ✓ **txn_data_master:** Table containing unique Request ID identifying the validation template.
- ✓ **txn_data:** Child Table of txn_data_master containing Request ID, Number Sequence and Ref Field Name to further specialize validation template.

Select the table in which you want to search for the txn data values. The Following Parameters are provided to user to search for existing txn data values.

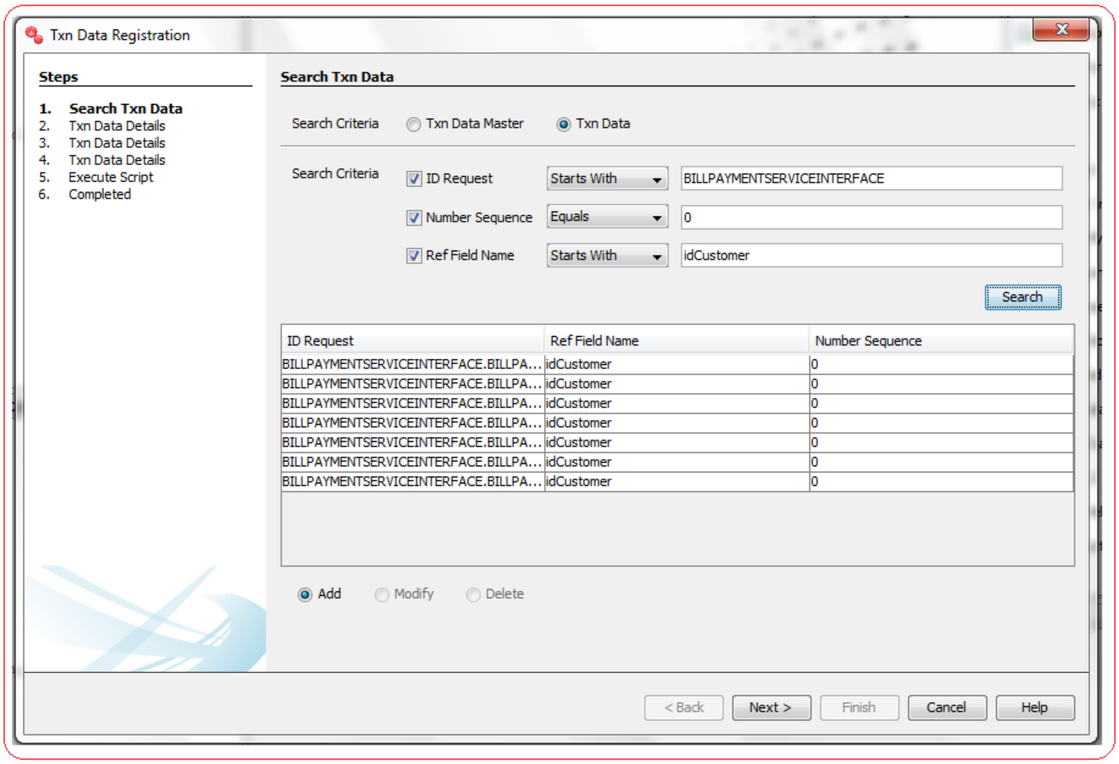
- ✓ txn_data_master
  - a) ID Request
- ✓ txn_data
  - a) ID Request
  - b) Number Sequence
  - c) Ref Field Name

Tick the parameters to search txn data with, select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available txn data values in the selected table matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.

txn_data_master – Search



txn_data – Search



Depending upon the search output one can do the following operations,

I. **Add New:** Select this radio button to add new txn data value in the selected table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

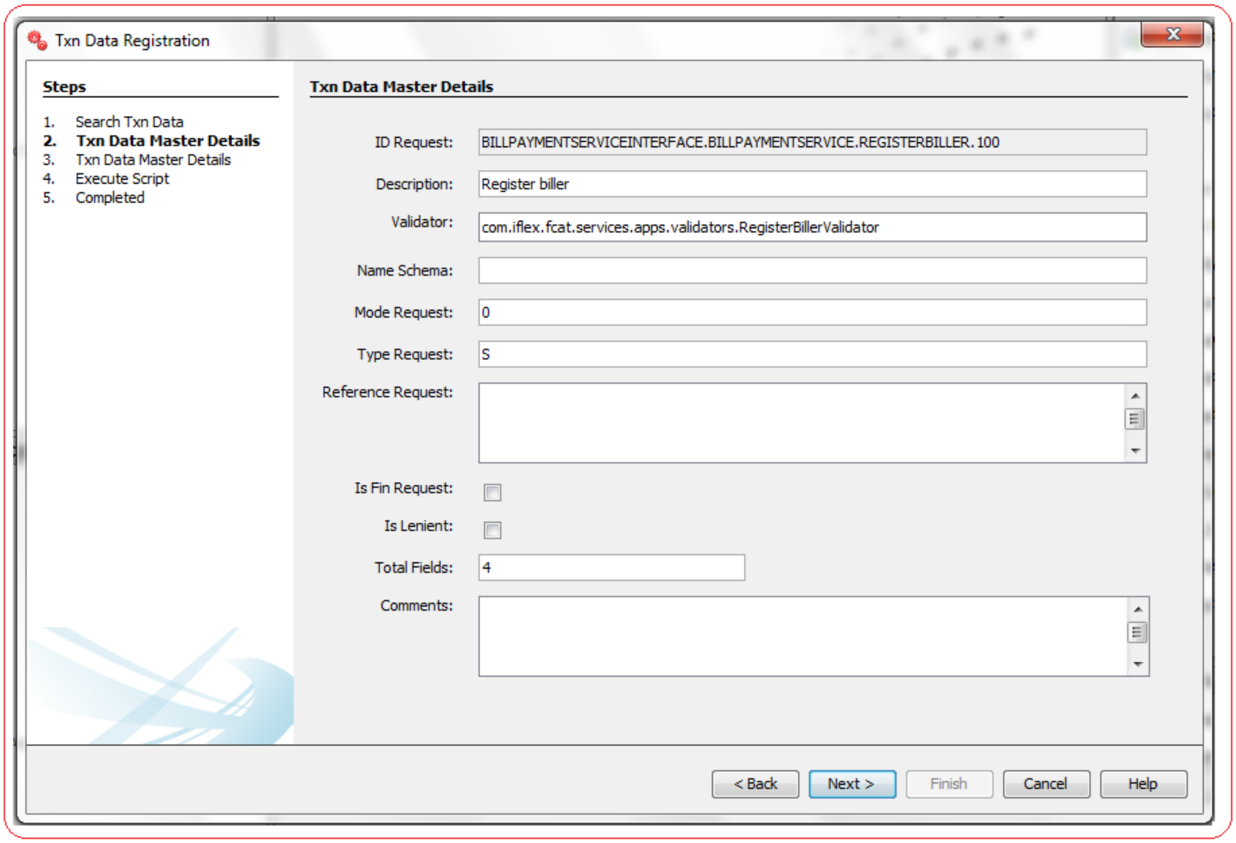
Click **Next** to go to next step.

Txn Data Registration – Edit

This step is meant to allow users to input details for a new txn data record in txn_data_master/txn_data, modify details of an existing one or view details of an existing one if user wants to delete the same.

txn_data_master

A panel showing/asking for the txn data master details will be displayed. Enter the details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the txn_data_master Table.

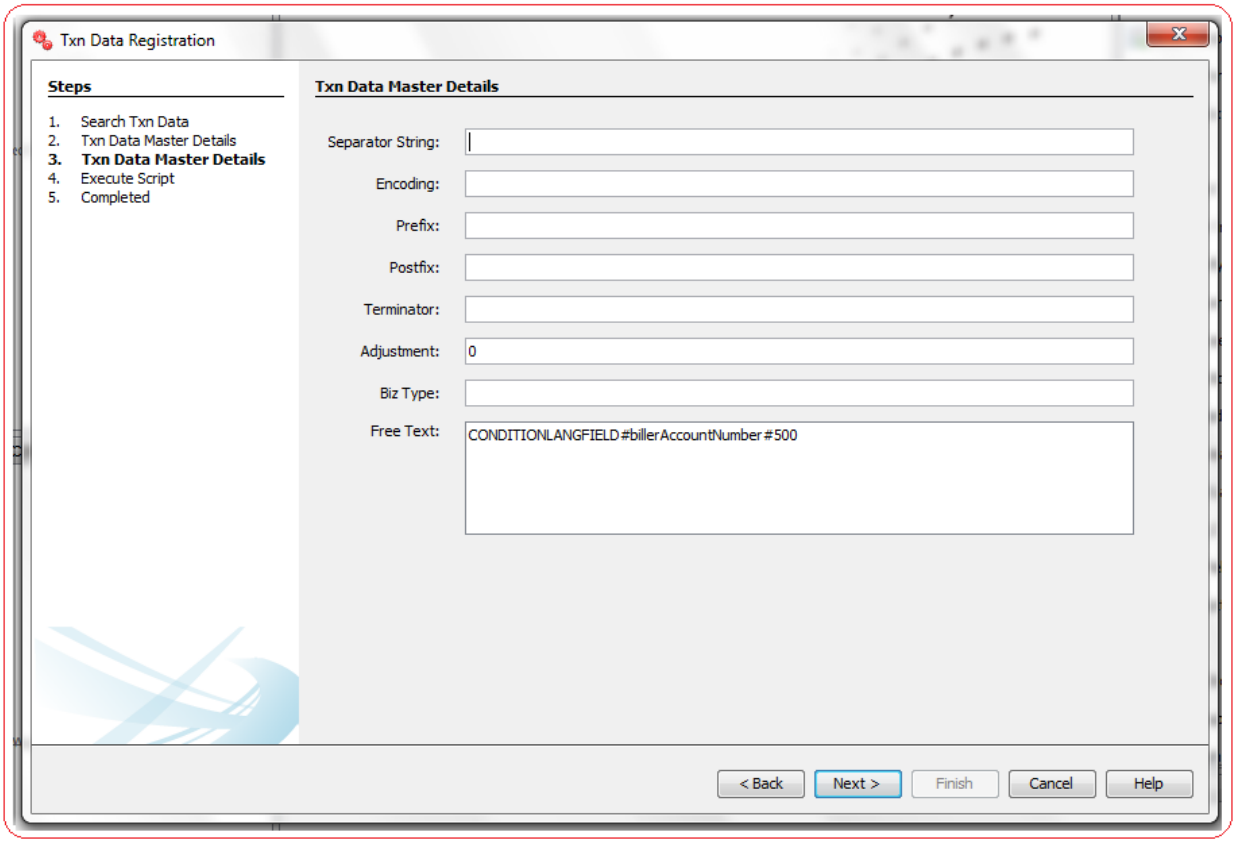


List of the User Input fields and their description are as follows:

- **ID Request:** Enter the Request ID of this validation template. It is Mandatory. The Request ID should be mapped to the ATS request definition in the MstAts table for the Validation Engine to invoke the validation against the Validation Template defined.
- **Description:** Enter the description of the transaction.
- **Validator:** Enter the Validator to be invoked for this txn data master value.
- **Name Schema:** Enter the Schema Name for this txn data master value.
- **Mode Request:** Enter the Request Mode for this txn data master value.
- **Type Request:** Enter the Request Type for this txn data master value.
- **Reference Request:** Enter the Reference Request for this txn data master value.
- **Is Fin Request:** Flag to indicate if this data value is fin request or not.
- **Is Lenient:** Flag to indicate if this data value is lenient or not.
- **Total Fields:** Enter the total fields for this txn data master value.
- **Comments:** Enter Comments if any for future reference.

Click **Next** to go to next step.

A second panel showing/asking for the txn_data_master details will be displayed. Enter the details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the txn_data_master table.



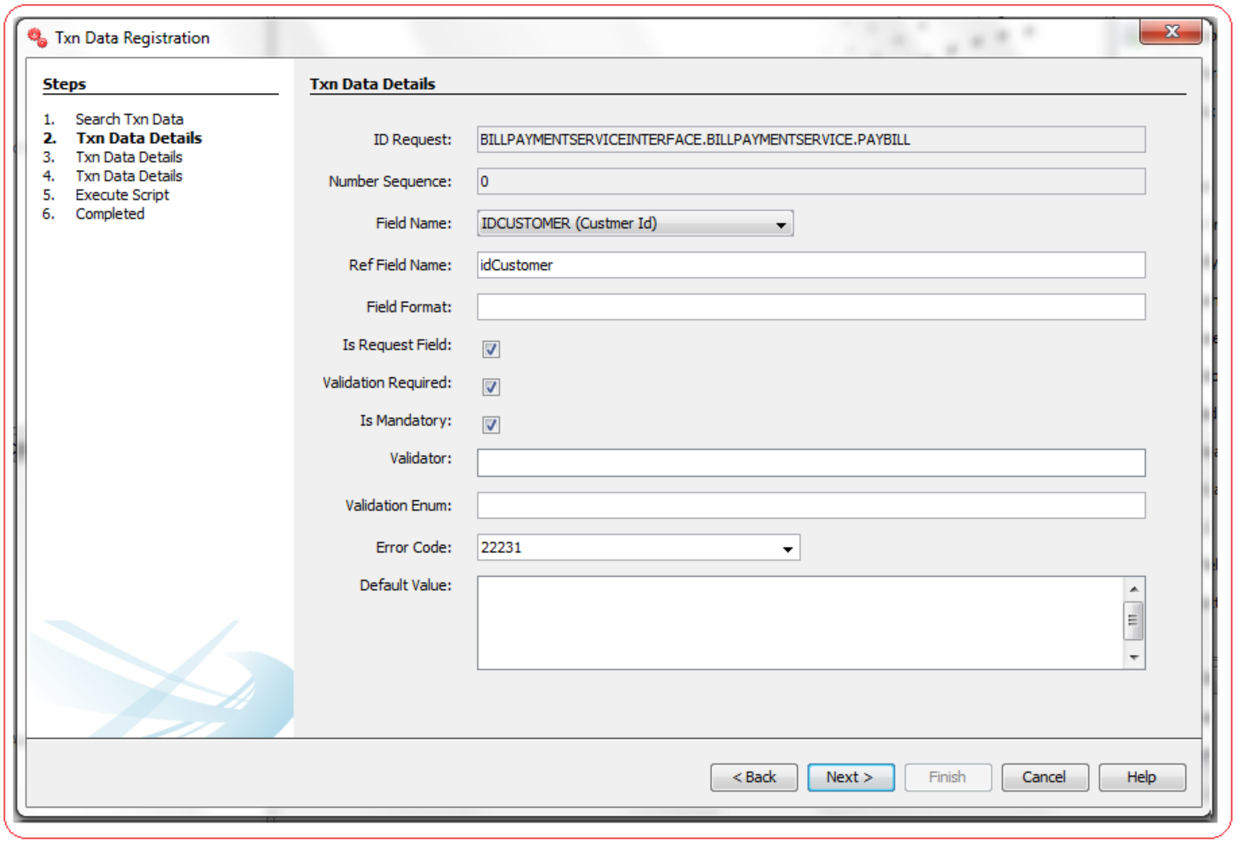
List of the User Input fields and their description are as follows:

- **Separator String:** Enter the separator string for this txn data master value.
- **Encoding:** Enter the encoding to be used for this txn data master value.
- **Prefix:** Enter Prefix string for this txn data master value.
- **Postfix:** Enter Postfix string for this txn data master value.
- **Terminator:** Enter terminator string for this txn data master value.
- **Adjustment:** Enter Adjustment string for this txn data master value.
- **Biz Type:** Enter the Business Type of this txn data master value.
- **Free Text:** Enter the free text to be used for this txn data master value.

Click **Next** to go to next step.

txn_data

A panel showing/asking for the txn_data details will be displayed. Enter the details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the txn_data table.

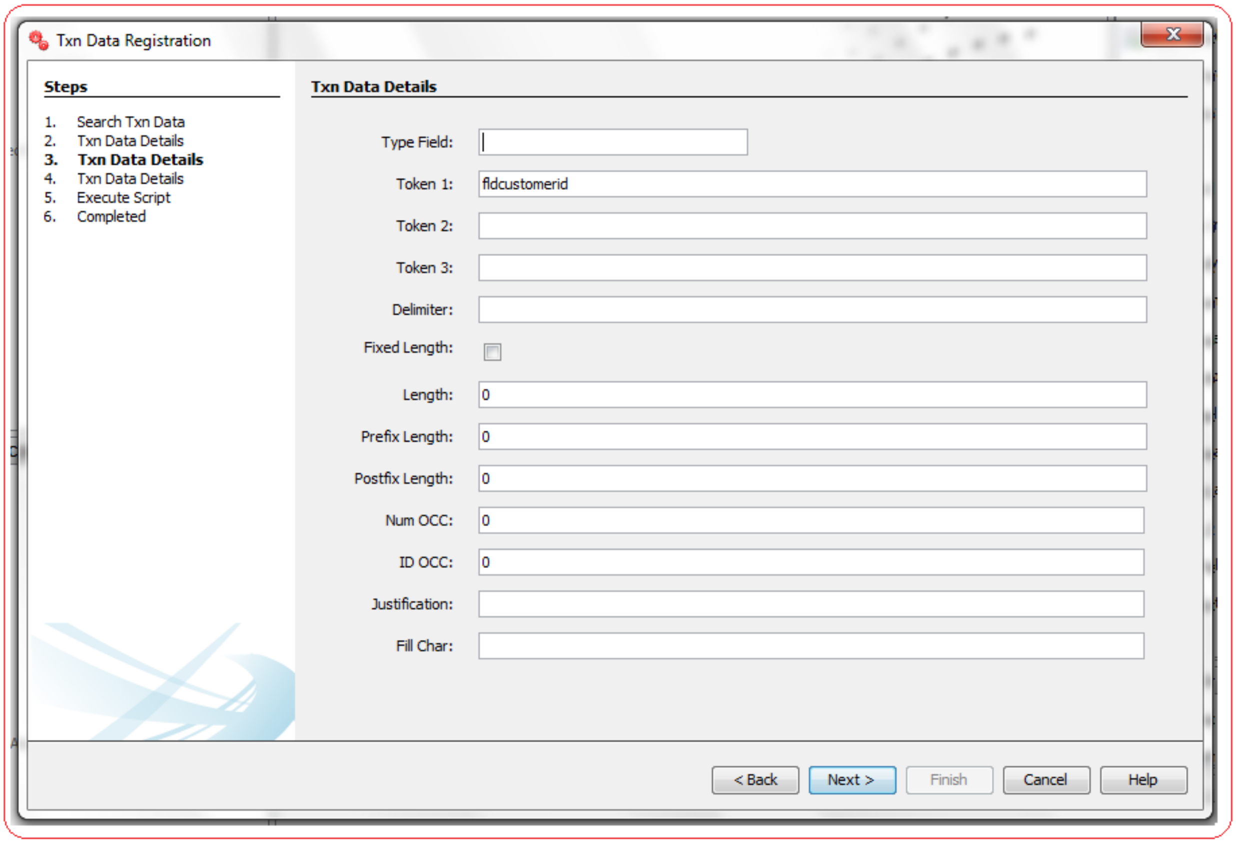


List of the User Input fields and their description are as follows:

- **ID Request:** Enter the Request ID for this txn data value.
- **Number Sequence:** Enter the Number sequence for this txn data value.
- **Field Name:** Enter the field name for this txn data value.
- **Ref Field Name:** Enter the reference field name for this txn data value.
- **Field Format:** Enter the field format for this txn data value.
- **Is Request Field:** Flag to indicate if this txn data value is request field or not.
- **Validation Required:** Flag to indicate if validation is required or not for this txn data value..
- **Is Mandatory:** Flag to indicate if this txn data value is mandatory or not.
- **Validator:** Enter the validator to be invoked for this txn data value.
- **Validation Enum:** Enter the Validation enumeration for this txn data value.
- **Error Code:** Enter the default value for this txn data value. Enter the Error Code of the application message to be displayed if validation fails.
- **Default Value:** Enter the default value for this txn data value.

Click **Next** to go to next step.

A second panel showing/asking for the txn_data details will be displayed. Enter the details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the txn_data table.

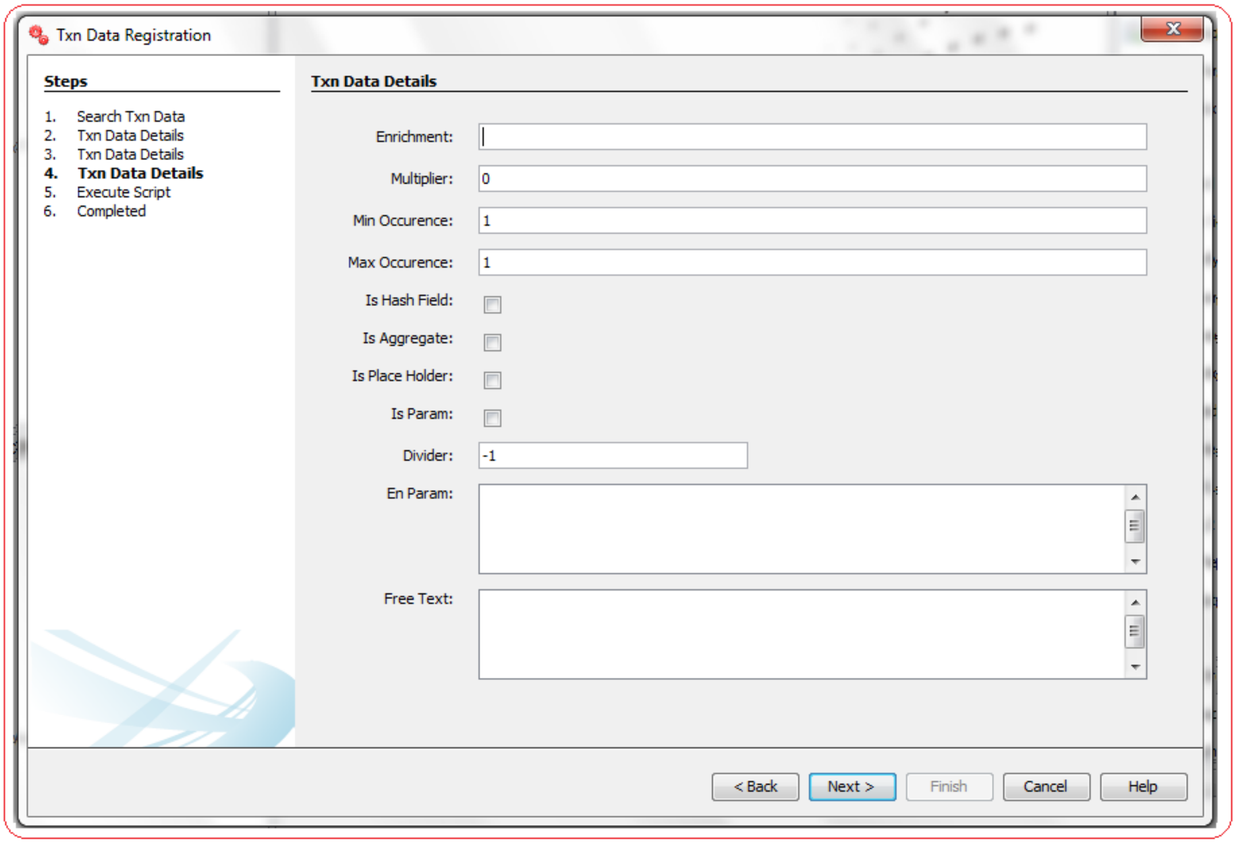


List of the User Input fields and their description are as follows:

- **Default Navigation:** Flag to indicate transaction should be made available as 'Default Navigation' for respective Entity, Usertype and Channel.
- **Type Field:** Enter the field type for this txn data value.
- **Token 1:** Enter token 1 to be used for this txn data value.
- **Token 2:** Enter token 2 to be used for this txn data value.
- **Token 3:** Enter token 3 to be used for this txn data value.
- **Delimiter:** Enter delimiter for this txn data value.
- **Fixed Length:** Enter fixed length of this txn data value.
- **Length:** Enter the length of this txn data value.
- **Prefix Length:** Enter the prefix length of this txn data value.
- **Postfix Length:** Enter the postfix length of this txn data value.
- **Num OCC:** Enter the number of occurrences for this txn data value.
- **ID OCC:** Enter the ID OCC field for this txn data value.
- **Justification:** Enter the justification for this txn data value.
- **Fill Char:** Enter the fill char for this txn data value.

Click **Next** to go to next step.

A third panel showing/asking for the txn_data details will be displayed. Enter the details if you are adding a new record, edit the details if you are modifying a record or Verify the details if you are deleting a record from the txn_data Table.



List of the User Input fields and their description are as follows:

- **Enrichment:** Enter the enrichment for this txn data value.
- **Multiplier:** Enter the Multiplier for this txn data value.
- **Min Occurrence:** Enter minimum number of occurrences for this txn data value.
- **Max Occurrence:** Enter the maximum number of occurrences for this txn data value.
- **Is Hash Field:** Flag to indicate if this txn data value is hash field or not.
- **Is Aggregate:** Flag to indicate if this txn data value is aggregate or not.
- **Is Place Holder:** Flag to indicate if this txn data value is place holder or not.
- **Is Param:** Flag to indicate if this txn data value is parameter or not.
- **Divider:** Enter the divider for this txn data value.
- **En Param:** Enter the enrichment parameter values for this txn data value.
- **Free Text:** Enter the free text for this txn data value.

Click **Next** to go to next step.

10. Txn Data Enrichment

Oracle FLEXCUBE Direct Banking Application houses all Data Enrichment properties required for configurations in the database table **txn_data_enrichment**.

txn_data_enrichment Definition

Column Name	Type	Nullable	Default	Comments
ENNAME	VARCHAR2 (255 CHAR)	No		Gives the unique name of the enrichment. Primary key of table.
ENTYPE	CHAR (1 CHAR)	No		It can be of two types. 1.Query Based: Define query in Query field. 2. Java Based: Enter Java class name in Java Class field.
ENNUMPARAMS	NUMBER (38)	No		Gives the number of parameters used in the query.
ENPARAMFIELDS	VARCHAR2 (2000 CHAR)	No		Provides the name of the data elements containing the parameter value. Multiple parameter fields can be separated with '#' character.
ENJAVACLASS	VARCHAR2 (255)	Yes		Value in this field is required if the Type is "Java Based". Link for Java Class is provided that gives the list of Java class.
ENREPLACEMAP	VARCHAR2 (255)	Yes		
ENQUERY	VARCHAR2 (255)	Yes		Value in this field is required if the Type is "Query Based". Link for Query is provided which gives the list of SQL query.
ENDESCRPTION	VARCHAR2 (255)	Yes		Gives the description of the enrichment.
ENJAVACLASSDESC	VARCHAR2 (200)	Yes		Gives the description of the Java Class.
LOADFLAG	CHAR (1 CHAR)	No	'D'	

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open Txn Data Enrichment Wizard, Go to Menu Options

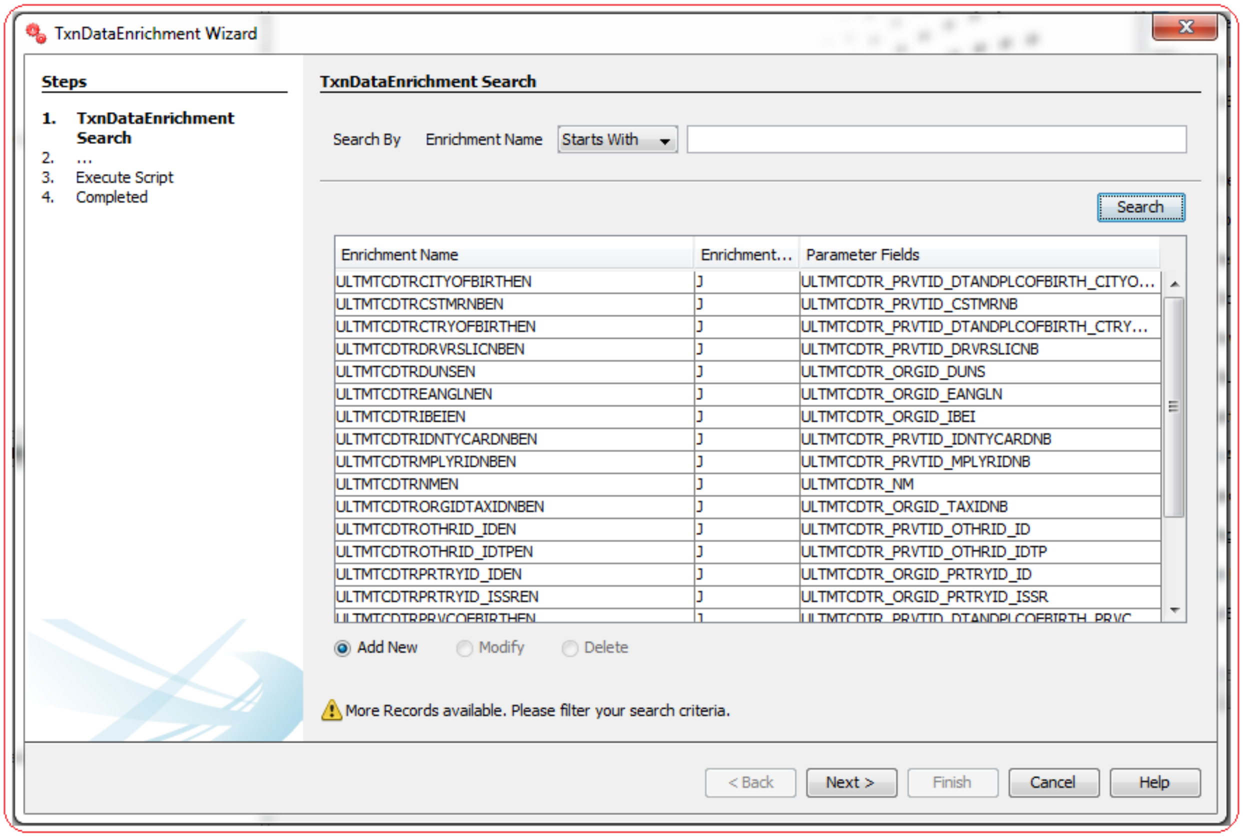
Tools → FCDB → Validation Engine → TxnDataEnrichment Wizard

TxnDataEnrichment - Search

This step is meant to allow the user to search for the existing entries available in txn_data_enrichment table of the connected Database. It is always advisable to search for existing entries before adding a new property in this table.

One can use the "Enrichment Name" parameter to filter the result described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available properties in the table **txn_data_enrichment** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



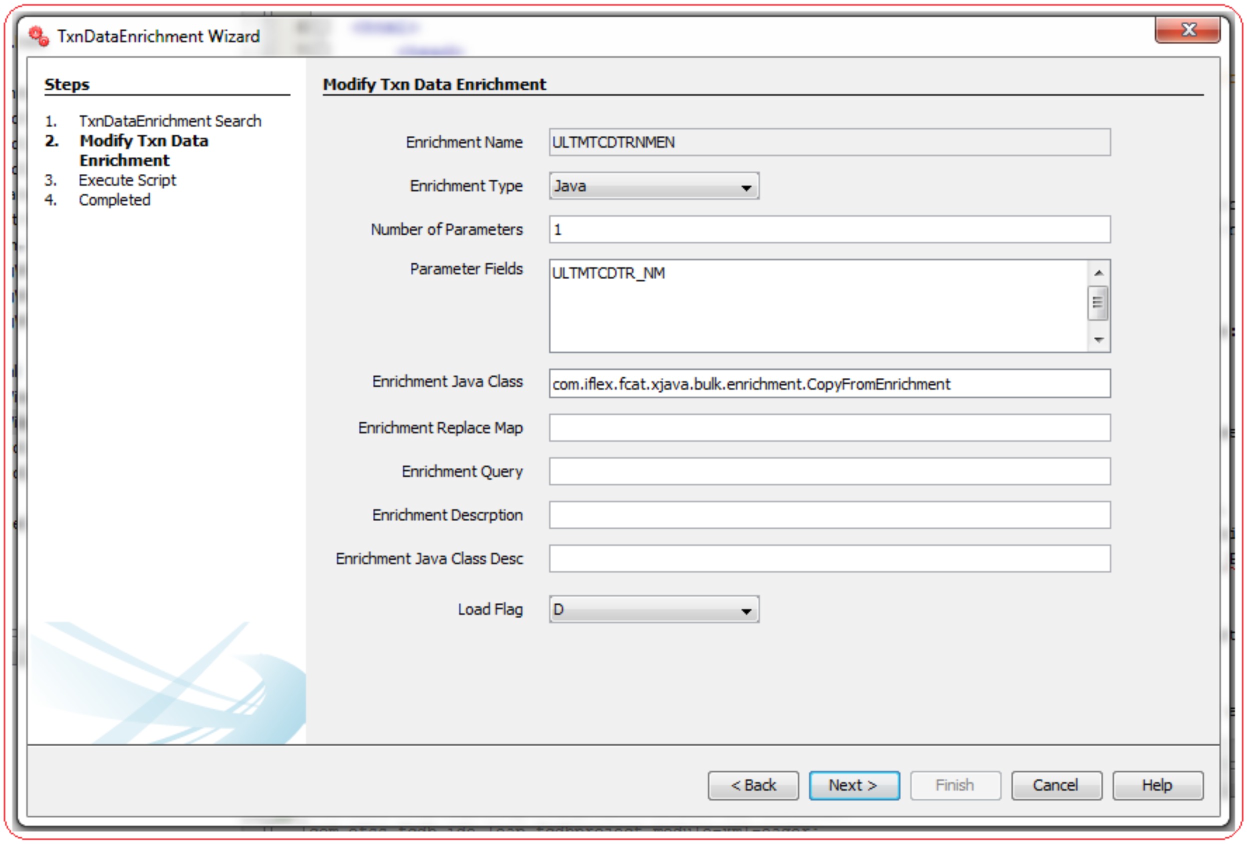
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new property in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

TxnDataEnrichment – Edit

This step is meant to allow users to input details for a new enrichment property, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Enrichment Name:** Enter the unique name of the enrichment.
- **Enrichment Type:** It can be of two types. Select one of the following:
  - ✓ Query Based: Define query in Query field.
  - ✓ Java Based: Enter Java class name in Java Class field.
- **Enrichment Num Params:** Enter the number of parameters used in the query.
- **Enrichment Param Fields:** Enter the name of the data elements containing the parameter value. Multiple parameter fields can be separated with '#' character.
- **Enrichment Java Class:** Value in this field is required if the Type is "Java Based". Link for Java Class is provided that gives the list of Java class.
- **Enrichment Replace Map:** Enter the map details for future use.
- **Enrichment Query:** Value in this field is required if the Type is "Query Based". Link for Query is provided which gives the list of SQL query
- **Enrichment Description:** Enter the description of the enrichment.
- **Enrichment Java Class Desc:** Enter the description of Java Class.
- **Load Flag:** Enter the Flag value. Default value is 'D'.

Click [Next](#) to go to next step.

11. MstQuery Registration

Oracle FLEXCUBE Direct Banking Application provides **MstQuery** table where developer has the option to configure database queries that can be invoked from FCDB Business Service java codes.

The **com.iflex.fcat.xjava.jdbc.JDBCEngine** class provides utility methods to execute queries on the database. Using the utility methods executeUpdate and executeQuery one can fire DB queries and get the result set. Developer has the option to either hardcode the queries while calling these utility methods or can configure the query in MstQuery table. Configuring Query in this table gives the privilege to the developer/implementation team to change the query without changing the java code.

MstQuery Definition

Column Name	Type	Nullable	Default	Comments
IDQUERY	VARCHAR2 (50 CHAR)	No		A unique identifier for the query. Primary key of table.
DATABASETYPE	VARCHAR2 (50 CHAR)	No	ORACLE	Database Type.
QUERYDESC	VARCHAR2 (255 CHAR)	No		Description for the query.
QUERYSTR	VARCHAR2 (4000 CHAR)	No		Query String.
STMTTYPE	VARCHAR2 (1 CHAR)	No	P	It can be of two types. 1.Prepared Statement (P) 2.Callable Statement (C)
QUERYTYPE	VARCHAR2 (1 CHAR)	No		It can be of following types. 1.Select (S) 2.Insert (I) 3.Update (U) 4.Delete (D) 5.Procedure/Function call (P)
MAXROWS	NUMBER (4,0)	No	0	Maximum number of rows query should return.
ISLDBQUERY	CHAR (1 CHAR)	No	Y	Flag to Identify whether query is for Local database or not.
NBRPARAMS	NUMBER(3,0)	No	0	Total number of parameters used in the query string.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open MstQuery Wizard, Go to Menu Options

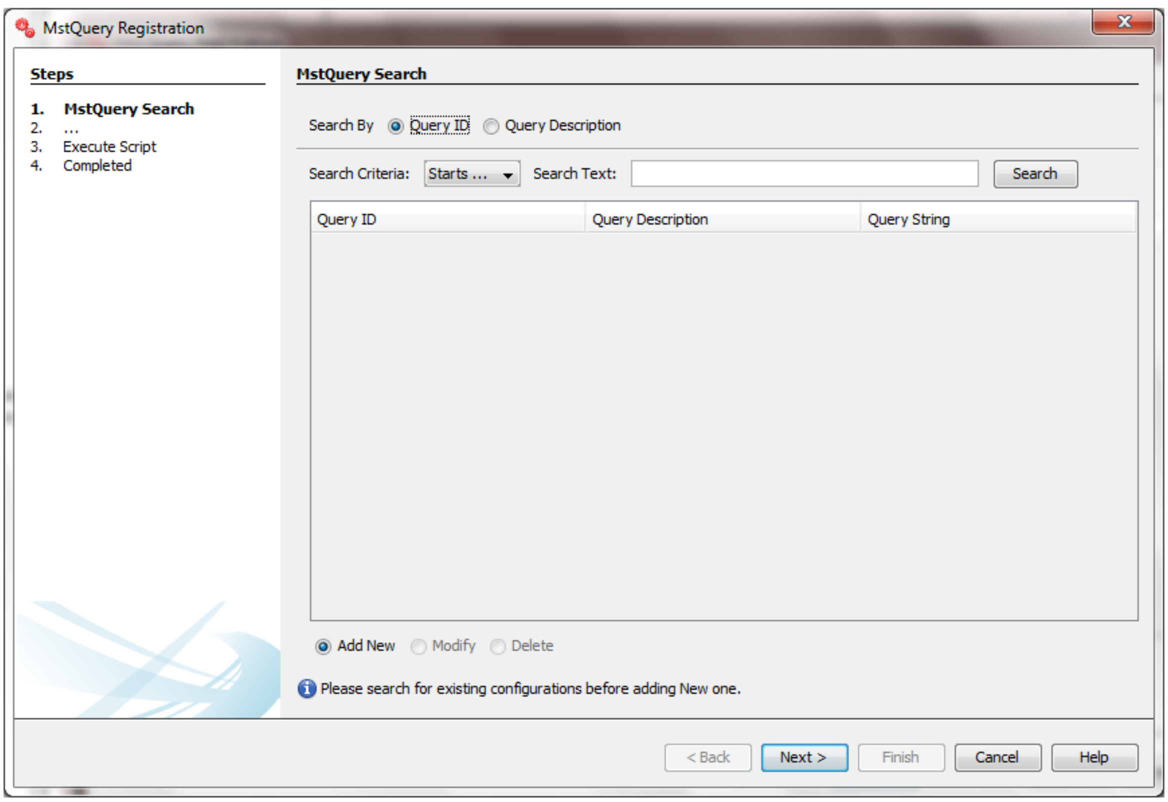
Tools → FCDB → Others → MstQuery Wizard.

MstQuery - Search

This step is meant to allow the user to search for the existing entries available in **MstQuery** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.

One can use the "Query Id" or "Query Description" parameter to filter the result described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table **MstQuery** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



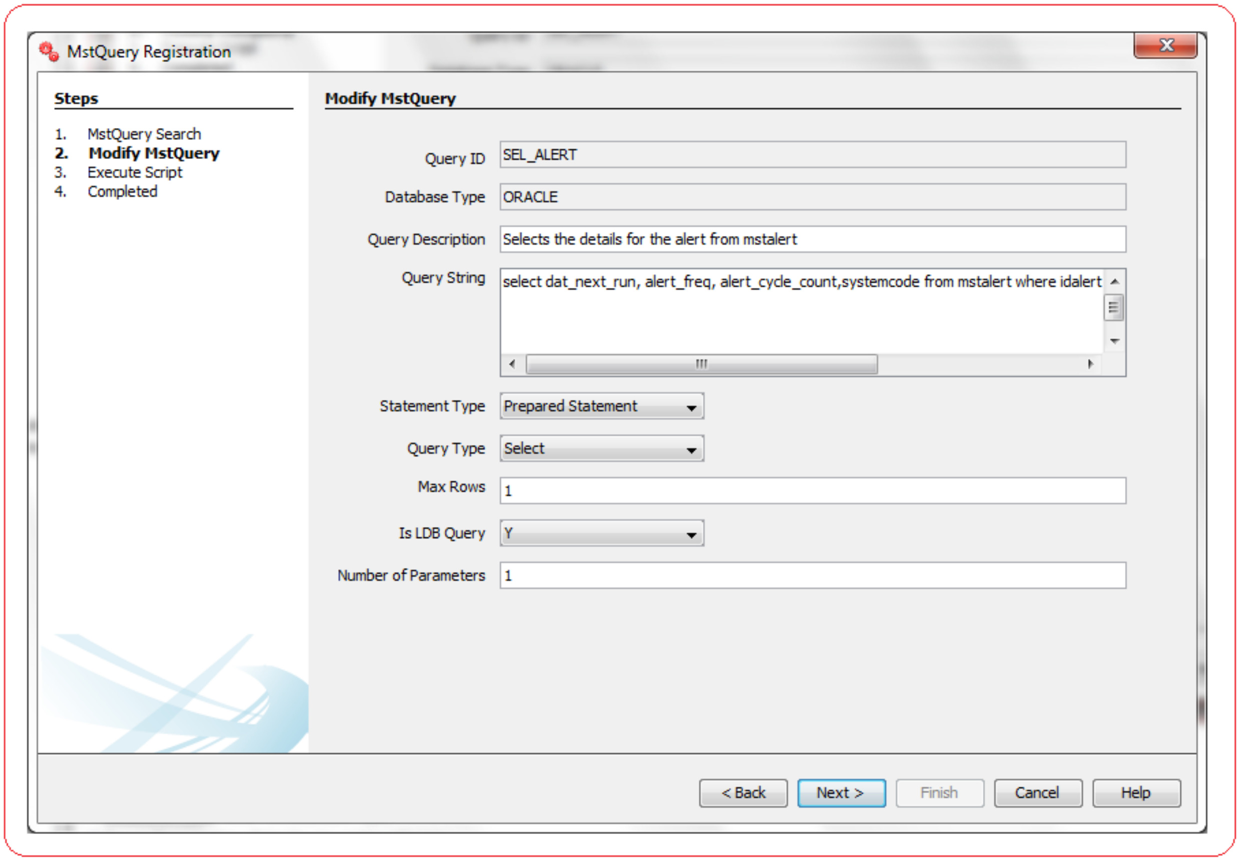
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstQuery - Edit

This step is meant to allow users to input details for a new MstQuery configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Query ID**: Enter a Unique Identifier for the Query.
- **Database Type**: Enter the Database Type.
- **Query Description**: Enter the description of query.
- **Query String**: Enter the Query String.
- **Statement Type**: Select the statement type.
- **Query Type**: Select the query type.
- **Max Rows**: Enter the maximum number of rows query should return.
- **Is LDB Query**: Select whether query is for local database or not.
- **Number of Parameters**: Enter the number of parameters used in the query.

Click **Next** to go to next step.

12. MstServicesMap Registration

Oracle FLEXCUBE Direct Banking Application provides **MstServicesMap** table where developer has the option to map service versions. Using this feature one can map one Version to another Version of the same FCDB Business Service.

Let's take an example. If one has registered a Business service in **mstservices** table with two different versions say 0 and 100 then using this mapping infrastructure one can force a particular input version (say 0) to always go through business implementation done for output version (say 100).

MstServicesMap Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		Entity ID.
TYPEUSER	VARCHAR2 (3 CHAR)	No		User Type.
NAMSERVICE	VARCHAR2 (255 CHAR)	No		Service Name for which mapping is defined.
INPUTVERSION	NUMBER	No		Service Input Version.
OUTPUTVERSION	NUMBER	No		Service Output Version.
ISENABLED	VARCHAR2 (1 CHAR)	No	Y	Identifies whether this service mapping is enabled or not.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open MstServicesMap Wizard, Go to Menu Options

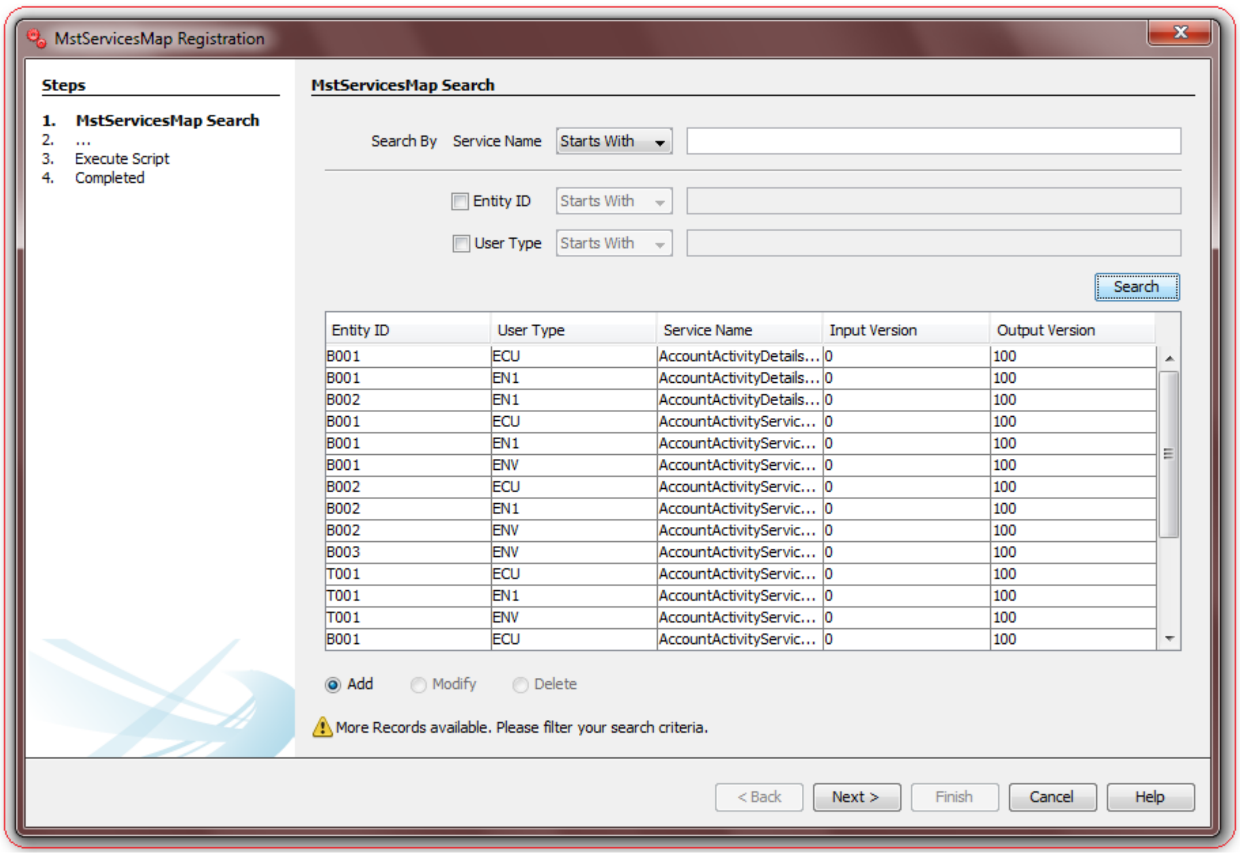
Tools → FCDB → Others → MstServicesMap Registration.

MstServicesMap - Search

This step is meant to allow the user to search for the existing entries available in **mstservicesmap** table of the connected Database. It is always advisable to search for existing entries before adding a new property in this table.

One can use the "Service Name" or "Entity ID" or "User Type" parameter to filter the result described below.

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available properties in the table mstservicesmap matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



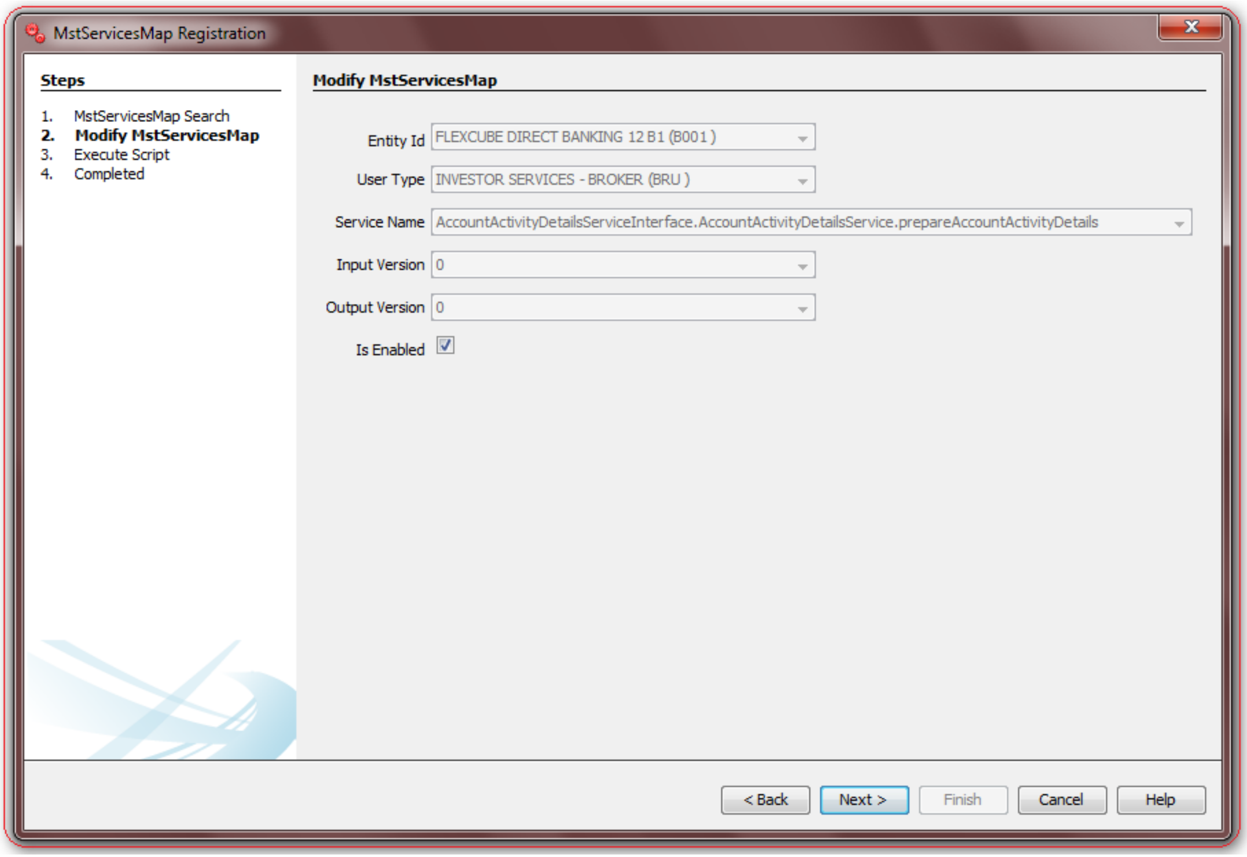
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstServicesMap - Edit

This step is meant to allow users to input details for a new MstServicesMap property, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Entity ID**: Select the Entity ID.
- **User Type**: Select the User Type.
- **Service Name**: Select the Service for which mapping needs to be done.
- **Input Version**: Select the Service Input Version.
- **Output Version**: Select the Service Output version.
- **Enabled**: Select the check box to enable this service mapping.

Click **Next** to go to next step.

13. MstUIDownload Registration

Oracle FLEXCUBE Direct Banking Application provides infrastructure using which one can configure pagination of grid data in a transaction, if required. This infrastructure provides user the option to select the list of columns to be displayed in the grid view and also download and print the data.

To use this infrastructure in a transaction one needs to do following configurations:

1. Create a table in your screen design and set table CSS as **"uidownload"**. This table will act as the container for UI Download infrastructure.
2. Set "Pagination Flag to 1 and Post Process Flag to 2" in **MstChannelATS** table.
3. Set "fldDataId and fldSectionId" **Hidden fields in Screen design**.
4. Use following two tables to do the necessary configuration for a particular transaction.

MstUIDownload Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		Entity ID.
TYPEUSER	VARCHAR2 (3 CHAR)	No		User Type.
IDTXN	VARCHAR2 (3 CHAR)	No		Transaction ID.
IDREQUEST	VARCHAR2 (50 CHAR)	No		Request ID.
IDCHANNEL	VARCHAR2 (2 CHAR)	No		Channel ID.
TYPEDOWNLOAD	VARCHAR2 (50 CHAR)	No		Download Type (PD/UD) – Predefined / User defined
TYPEFORMAT	VARCHAR2 (50 CHAR)	No		Comma separated list of format for the download (PDF, XSL, HTML, RTF).
PATH	VARCHAR2 (255 CHAR)	Yes		XPath for the DTO (e.g. "/faml/response/.. /XYZDTO" etc)
RECORDSPERPAGE	Number	Yes		Page Size (No. of records to fetch at one go)
SORTCOLUMN	VARCHAR2 (255 CHAR)	Yes		Comma separated list of Column number on which sorting is needed
SORTORDER	VARCHAR2 (255 CHAR)	Yes		Sorting Order (Ascending (A)/Descending (D)), Comma separated list for multiple columns.
ADTNL_PARAMS	VARCHAR2 (4000 CHAR)	Yes		Column for Additional Parameters.

MstUIDownloadParams Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		Entity ID.
TYPEUSER	VARCHAR2 (3 CHAR)	No		User Type.
IDTXN	VARCHAR2 (3 CHAR)	No		Transaction ID.
IDPARAM	Number	No		ID for Parameter.
IDREQUEST	VARCHAR2 (50 CHAR)	No		Request ID.
IDCHANNEL	VARCHAR2 (2 CHAR)	No		Channel ID.
NAMPARAM	VARCHAR2 (50 CHAR)	No		Parameter Name (Maps to DATAVALUE of APPLDATA table e.g. K_INTEREST_RATE).
PARAMSEQ	NUMBER	No		Parameter Sequence order to display the columns.
TYPEPARAM	VARCHAR2 (2 CHAR)	No	'S'	Parameter type (Default is ‘S’) A=Amount, N=Number, S=String, D=Date, B=Boolean, I=Integer, P=Percentage, T=Timestamp, O=Other.
ISENABLED	VARCHAR2 (1 CHAR)	No	'Y'	Flag to identify whether column will be displayed or not.
ISFIXED	VARCHAR2 (1 CHAR)	No	'Y'	Flag to identify whether column will be default visible or not.
NAMPATH	VARCHAR2 (500 CHAR)	Yes		Path to the parameter Relative from the node specified at MSTUIDOWNLOAD.PATH.
ISLINK	VARCHAR2 (1 CHAR)	No	'N'	Flag to identify whether the column is hyperlink.
NAMFUNCTION	VARCHAR2 (255 CHAR)	Yes		Name of the JavaScript function to be called on click.
ARGFUNCTION	VARCHAR2 (255 CHAR)	Yes		Arguments of the function (IDPARAMs to pass as arguments to the NAMFUNCTION, separated by ~^. For example - 0~^1~^2.
ALIGN	VARCHAR2 (1 CHAR)	No	'L'	Alignment in the cell L=Left, R=Right
FIELDLENGTH	NUMBER	Yes		Length of the field, applicable for TEXT file download
DYNAMICNAMPATH	VARCHAR2 (255 CHAR)	Yes	'*'	Used in places where the actual path is not "nampath". For e.g. in Auth, the response is Auth specific DTO but it is just a container to store different types of DTOs.
REFIDPARAM	VARCHAR2 (100 CHAR)	Yes		Referenced IDPARAM which means in conjunction with this field which field is related – e.g. along with amount field Currency field is to be displayed.
ADTNL_PARAMS	VARCHAR2 (4000 CHAR)	Yes		Column for Additional Parameters.

TYPEFIELD	CHAR (1 CHAR)	No		'S' or 'R', 'R' to be used if this field needs to be displayed as search criteria in the downloaded file.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open MstUIDownload Wizard, Go to Menu Options

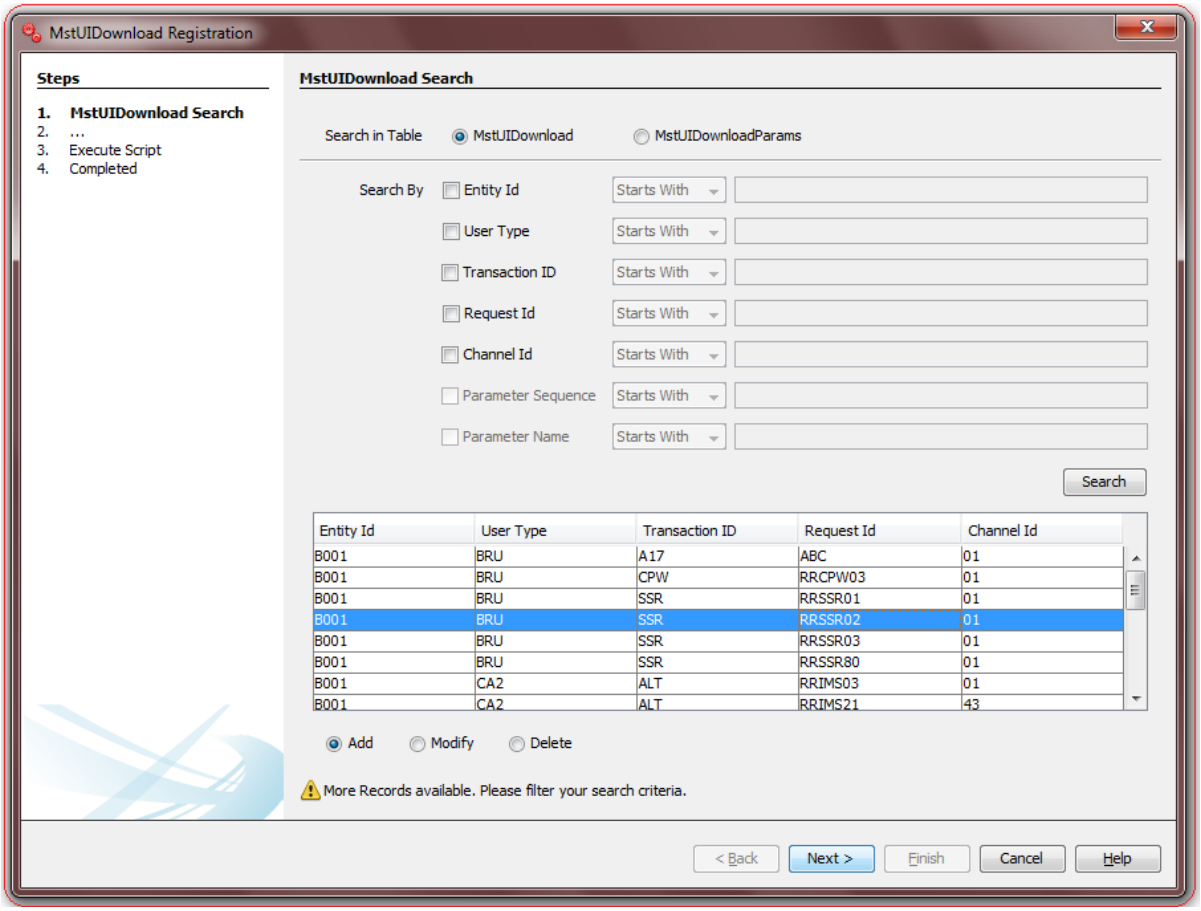
Tools → FCDB → Others → MstUIDownload Registration.

MstUIDownload Registration – Search

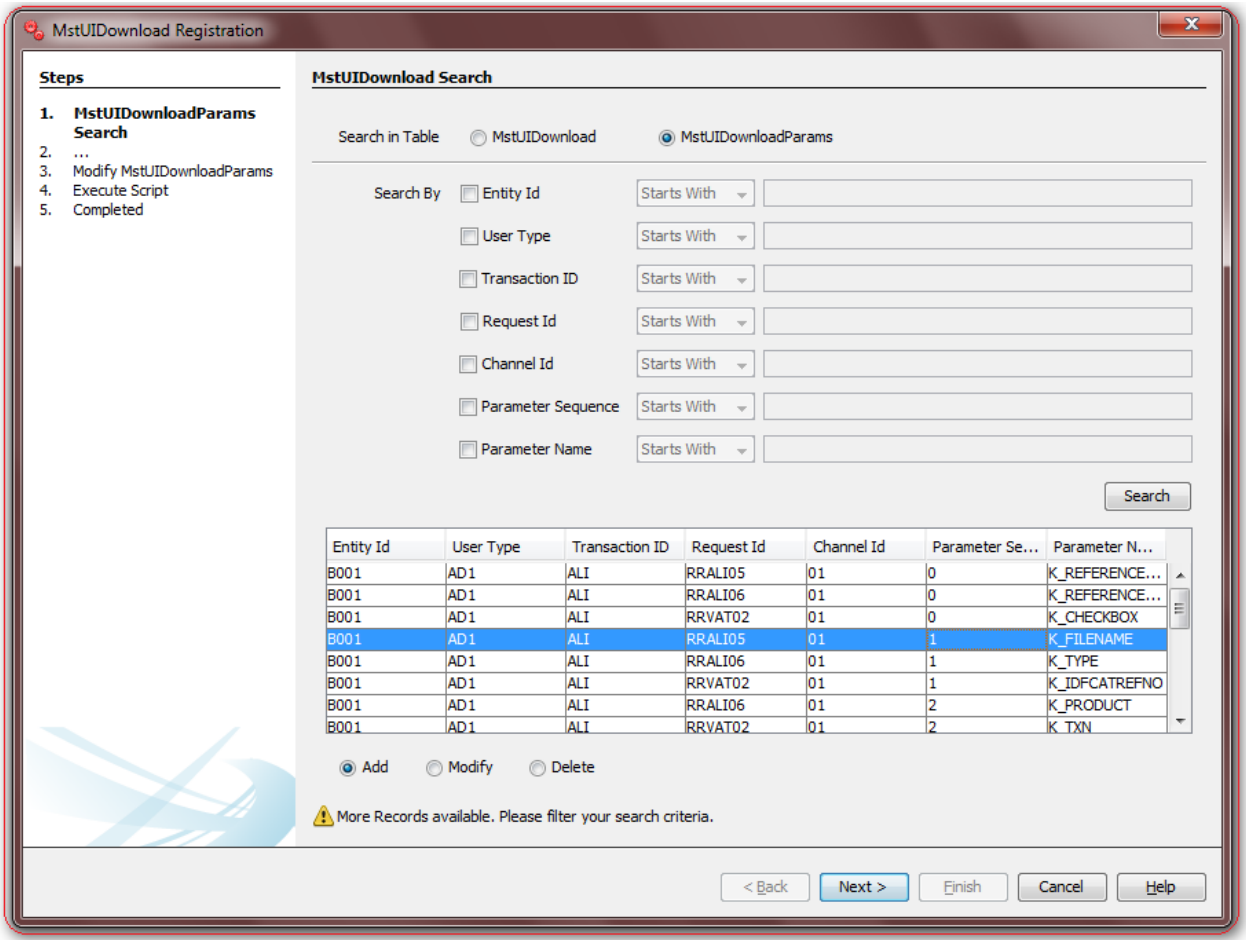
This step is meant to allow the user to search for the existing entries in **mstuidownload/mstuidownloadparams** table of the connected Database. It is always advisable to search for existing entries before adding a new property in this table.

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available properties in the table **mstuidownload/mstuidownloadparams** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.

MstUIDownload - Search



MstUIDownloadParams - Search



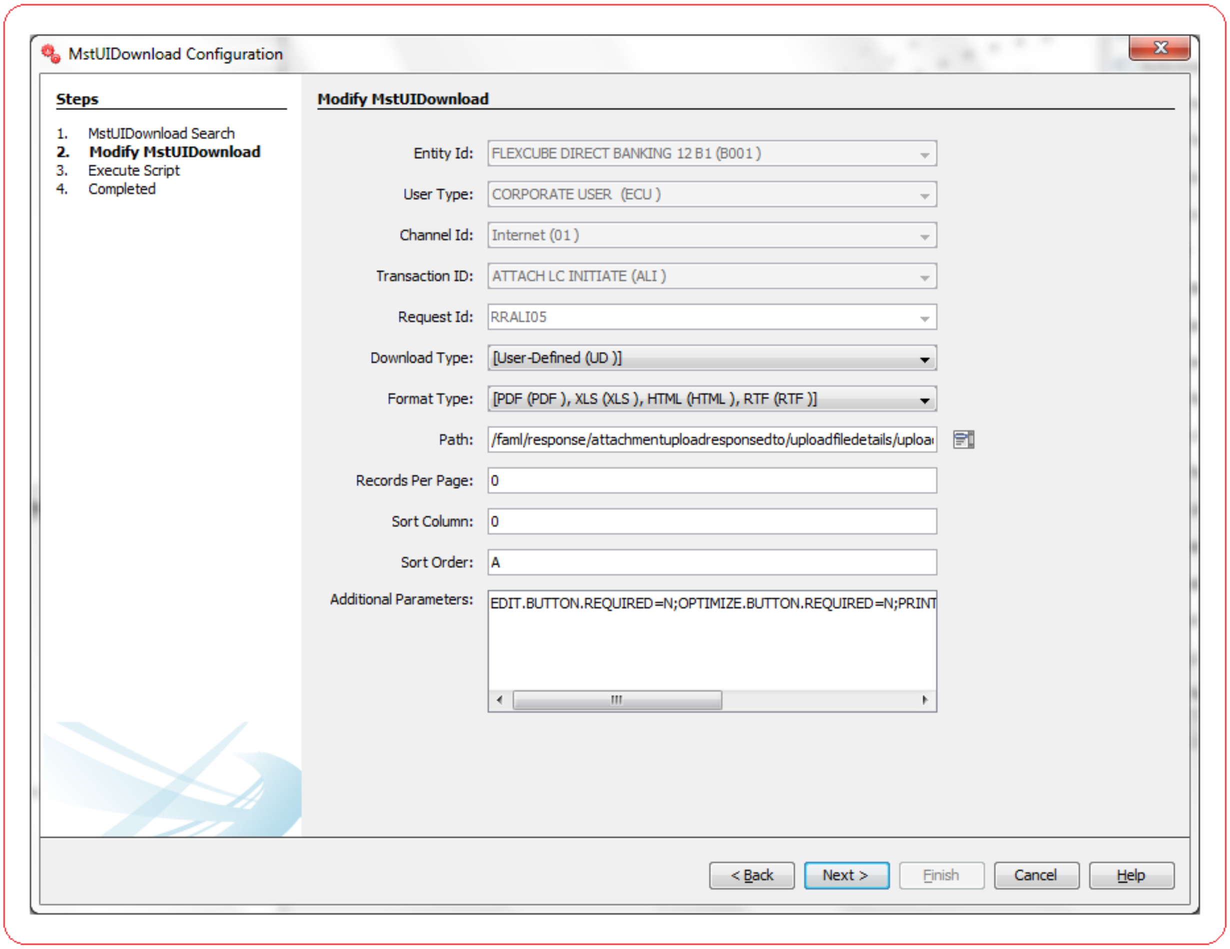
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstUIDownload - Edit

This step is meant to allow users to input details for a new **mstuidownload/mstuidownloadparams** property, modify details of an existing one or view details of an existing one if user wants to delete the same.

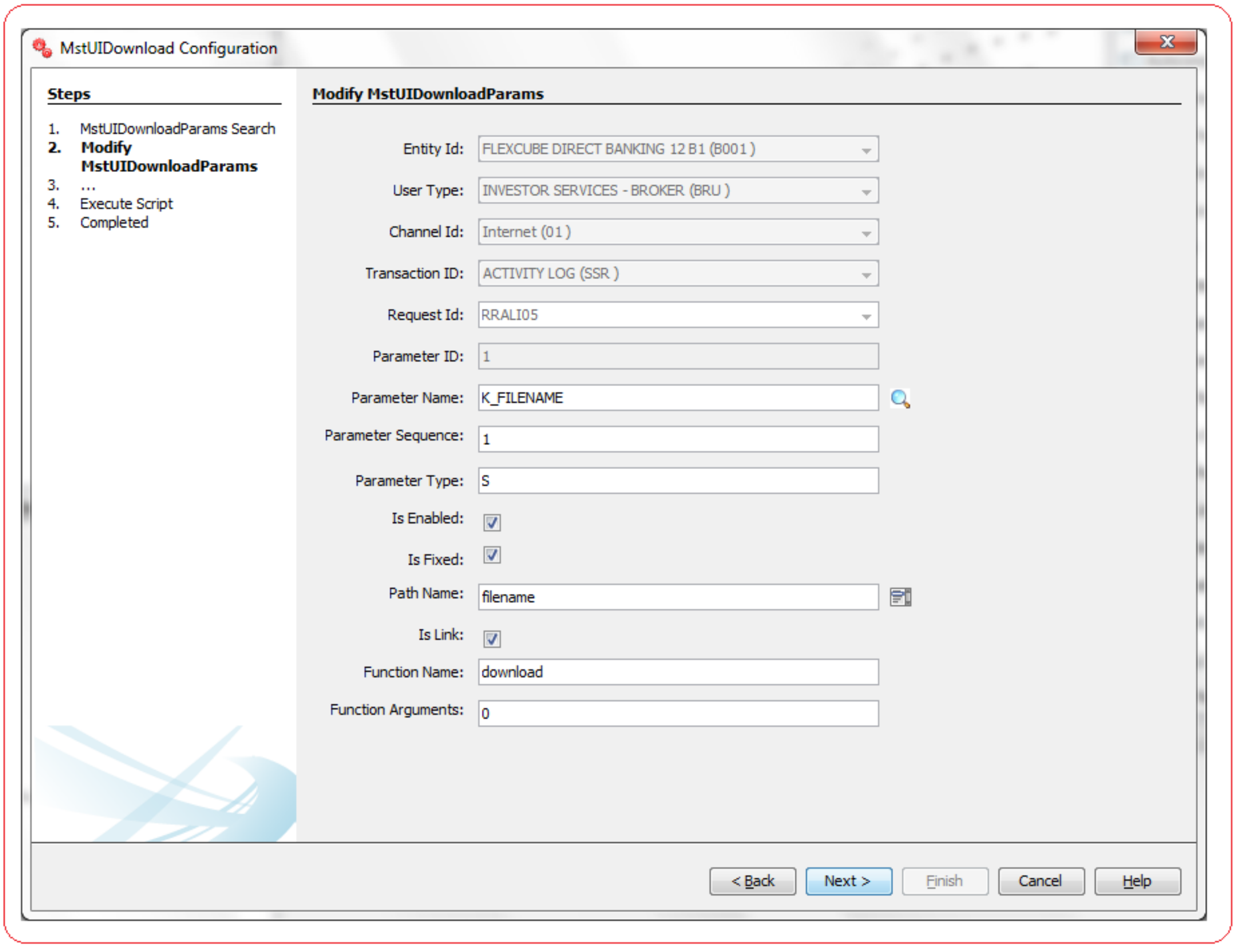


List of the User Input fields and their description are as follows:

- **Entity ID:** Select the Five letter alphanumeric value to uniquely identify Entity.
- **User Type:** Select the Three letter alphanumeric value to uniquely identify User Type.
- **Channel ID:** Select the Two letter numeric value to uniquely identify Channel.
- **Transaction ID:** Select the Three letter alphanumeric value to uniquely identify this transaction.
- **Request ID:** Select the Alphanumeric value to uniquely identify Request.
- **Download Type:** Select the Download Type (PD/UD) – Predefined / User defined.
- **Format Type:** Select the Comma separated list of format for the download (PDF, XLS, HTML, RTF).
- **Path:** Enter the XPath for the DTO (e.g. "/faml/response/.. /XYZDTO" etc).
- **Records Per Page:** Enter the Page Size (No. of records to fetch at one go).
- **Sort Column:** Enter the Comma separated list of Column number on which sorting is needed.
- **Sort Order:** Enter the Sorting Order (Ascending(A)/Descending(D)), Comma separated list for multiple columns.
- **Additional Parameters:** Enter the Column for Additional Parameters.

Click **Next** to go to next step.

MstUIDownloadParams - Edit

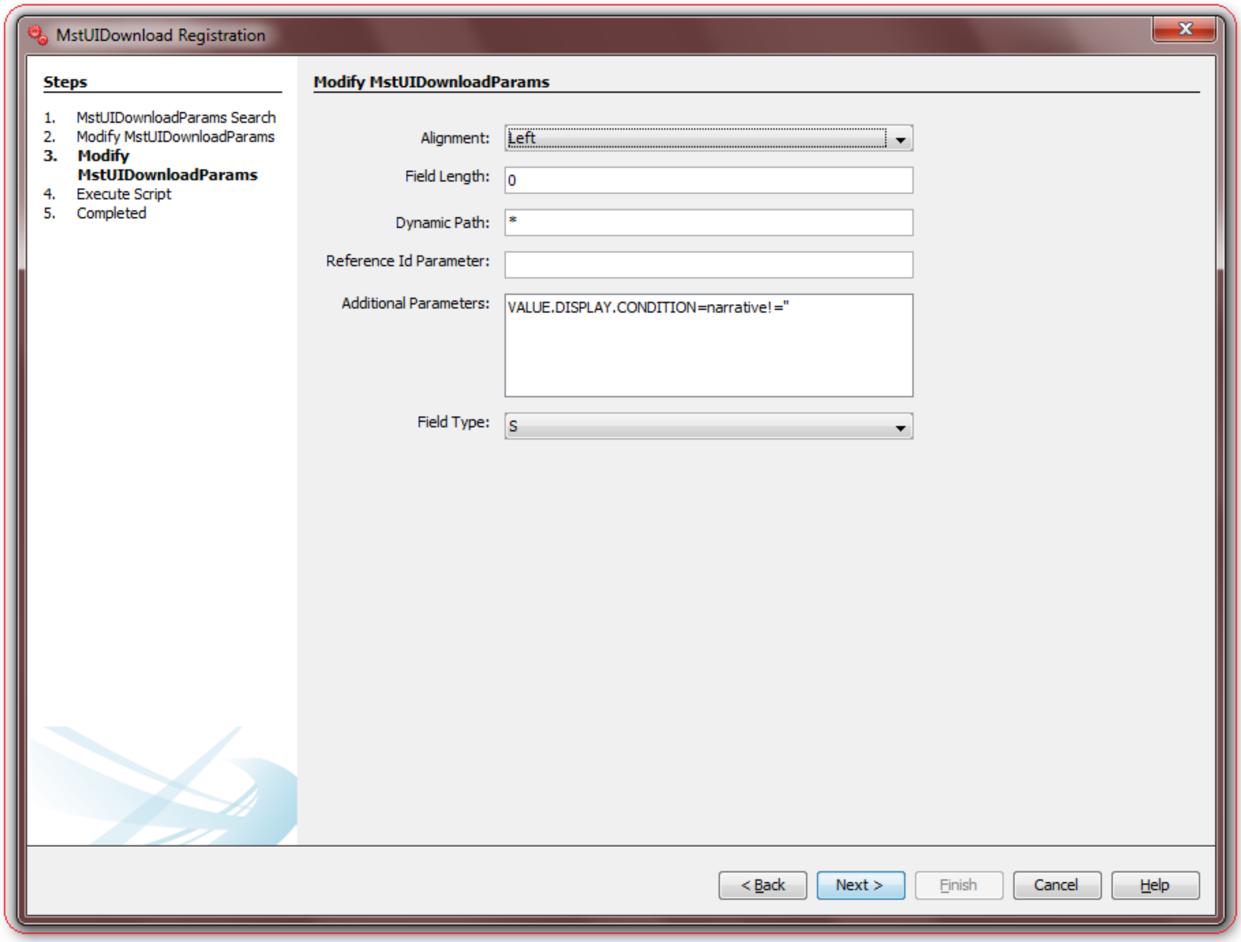


List of the User Input fields and their description are as follows:

- **Entity ID:** Select the Five letter alphanumeric value to uniquely identify Entity.
- **User Type:** Select the Three letter alphanumeric value to uniquely identify User Type.
- **Channel ID:** Select the Two letter numeric value to uniquely identify Channel.
- **Transaction ID:** Select the Three letter alphanumeric value to uniquely identify this transaction.
- **Request ID:** Select the Alphanumeric value to uniquely identify Request.
- **Parameter ID:** Enter the ID for the Parameter.
- **Parameter Name:** Enter the Parameter Name (Maps to DATAVALUE of APPLDATA table e.g. K_INTEREST_RATE).
- **Parameter Sequence:** Enter the Parameter Sequence order to display the columns.
- **Parameter Type:** Enter the Parameter type (Default is 'S') A=Amount, N=Number, S=String, D=Date, B=Boolean, I=Integer, P=Percentage, T=Timestamp, O=Other.
- **Is Enabled:** Flag to identify whether column will be displayed or not.
- **Is Fixed:** Flag to identify whether column will be default visible or not.
- **Path Name:** Enter the Path to the parameter Relative from the node specified at MSTUIDOWNLOAD.PATH.
- **Is Link:** Flag to identify whether the column is hyperlink.

- **Function Name:** Enter the Name of the JavaScript function to be called on click.
- **Function Arguments:** Enter the Arguments of the function (IDPARAMs to pass as arguments to the NAMFUNCTION, separated by ~^. For example - 0~^1~^2.

Click **Next** to go to next step.



List of the User Input fields and their description are as follows:

- **Align:** Select the Alignment in the cell L=Left, R=Right.
- **Field Length:** Enter the Length of the field, applicable for TEXT file download.
- **Dynamic Path:** Enter the Used in places where the actual path is not "name path". For e.g. in Auth, the response is Auth specific DTO but it is just a container to store different types of DTOs.
- **Reference ID Parameter:** Enter the Referenced IDPARAM which means in conjunction with this field which field is related – e.g. along with amount field Currency field is to be displayed.
- **Additional Parameters:** Enter the Column for Additional Parameters.
- **Field Type:** Select the Field Type. 'S' or 'R', 'R' to be used if this field needs to be displayed as search criteria in the downloaded file.

Click **Next** to go to next step.

14. MstUserAgentTxn Registration

Oracle FLEXCUBE Direct Banking Application houses the table **mstuseragenttxn** where developer can configure the menu options for transactions in Mobile Banking using Android Phone/Tablet and iPhone/iPad Apps. This is an additional table to override the menu configurations done in **mstusertypetxn**.

MstUserAgentTxn Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		A unique identifier for the entity. Primary key of table.
USERTYPE	VARCHAR2 (100 CHAR)	No		Description of the entity.
IDCHANNEL	CHAR (1 CHAR)	No		Entity Type.
IDUSERAGENT	VARCHAR2 (5 CHAR)	Yes		Parent Id of entity.
IDTXN	NUMBER	No		Sort Id of entity.
TOKEN1	VARCHAR2 (30 CHAR)	Yes		
TOKEN2	VARCHAR2 (30 CHAR)	Yes		Main menu under which this transaction will come, description of main menus is defined in appldata table with dataname = 'MENU_TAB_DESC' and datavalue = <value of this column>.
TOKEN3	VARCHAR2 (30 CHAR)	Yes		
TOKEN4	VARCHAR2 (30 CHAR)	Yes		
TOKEN5	VARCHAR2 (30 CHAR)	Yes		Numeric value to define the position of transaction in the transaction group.
TXNGROUP	VARCHAR2 (10 CHAR)	Yes		Group under which this transaction will come, description of group is defined in appldata table with dataname = 'TXN_GROUP_DESC' and datavalue = <Value of this column>.
INITREQUESTID	VARCHAR2 (8 CHAR)	Yes		Registered work flow step id (refer to MstChannelATS table) getting called on click of this transaction from menu. If no value specified first step will be called.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open Entity Registration Wizard, Go to Menu Options

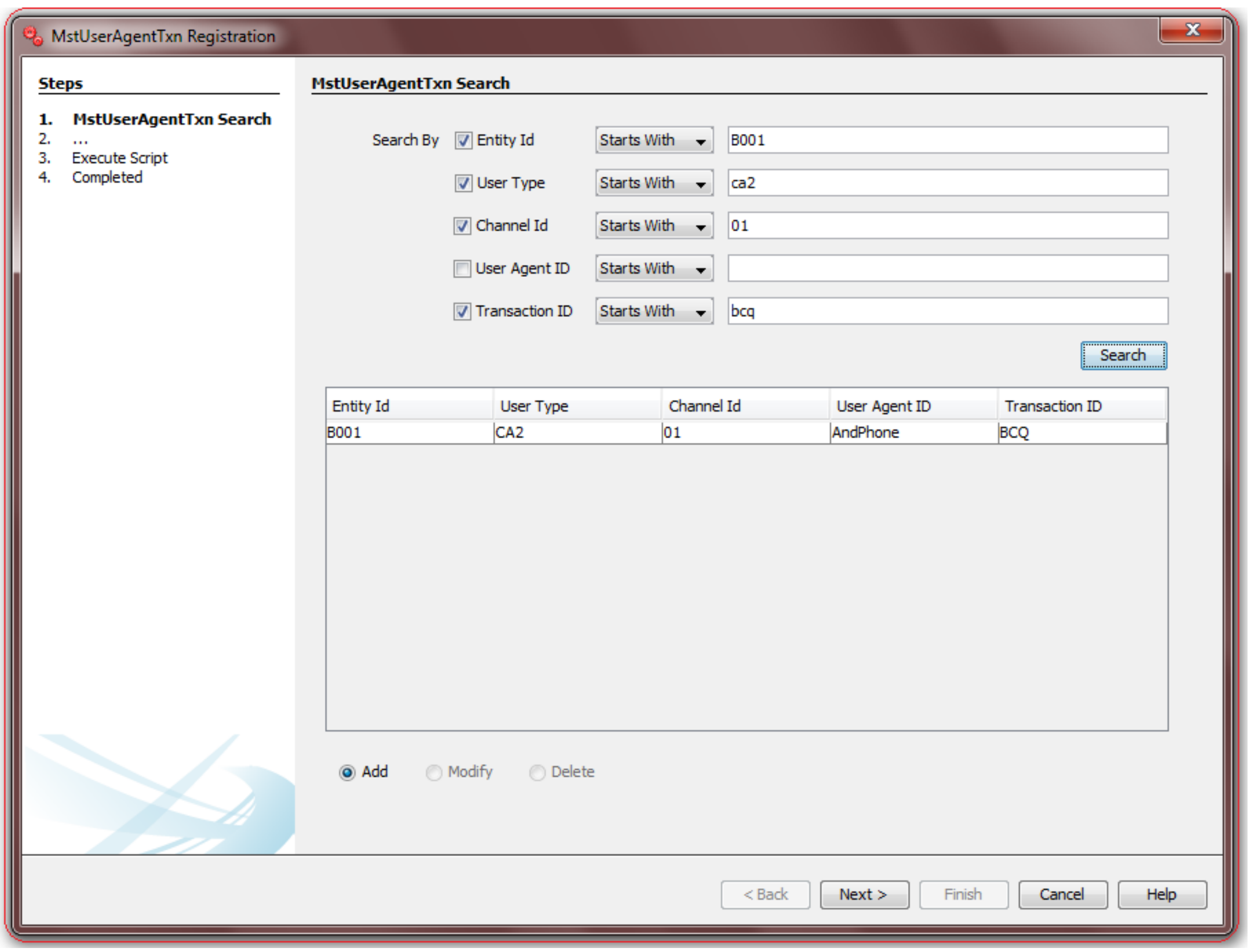
Tools → FCDB → Others → MstUserAgentTxn Registration.

MstUserAgentTxn - Search

This step is meant to allow the user to search for the existing entries available in **mstuseragenttxn** table of the connected Database. It is always advisable to search for existing entries before adding a new property in this table.

One can use the "Entity ID" or "User Type" or "Channel ID" or "User Agent ID" or "Transaction ID" parameter to filter the result described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available properties in the table **mstuseragenttxn** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.

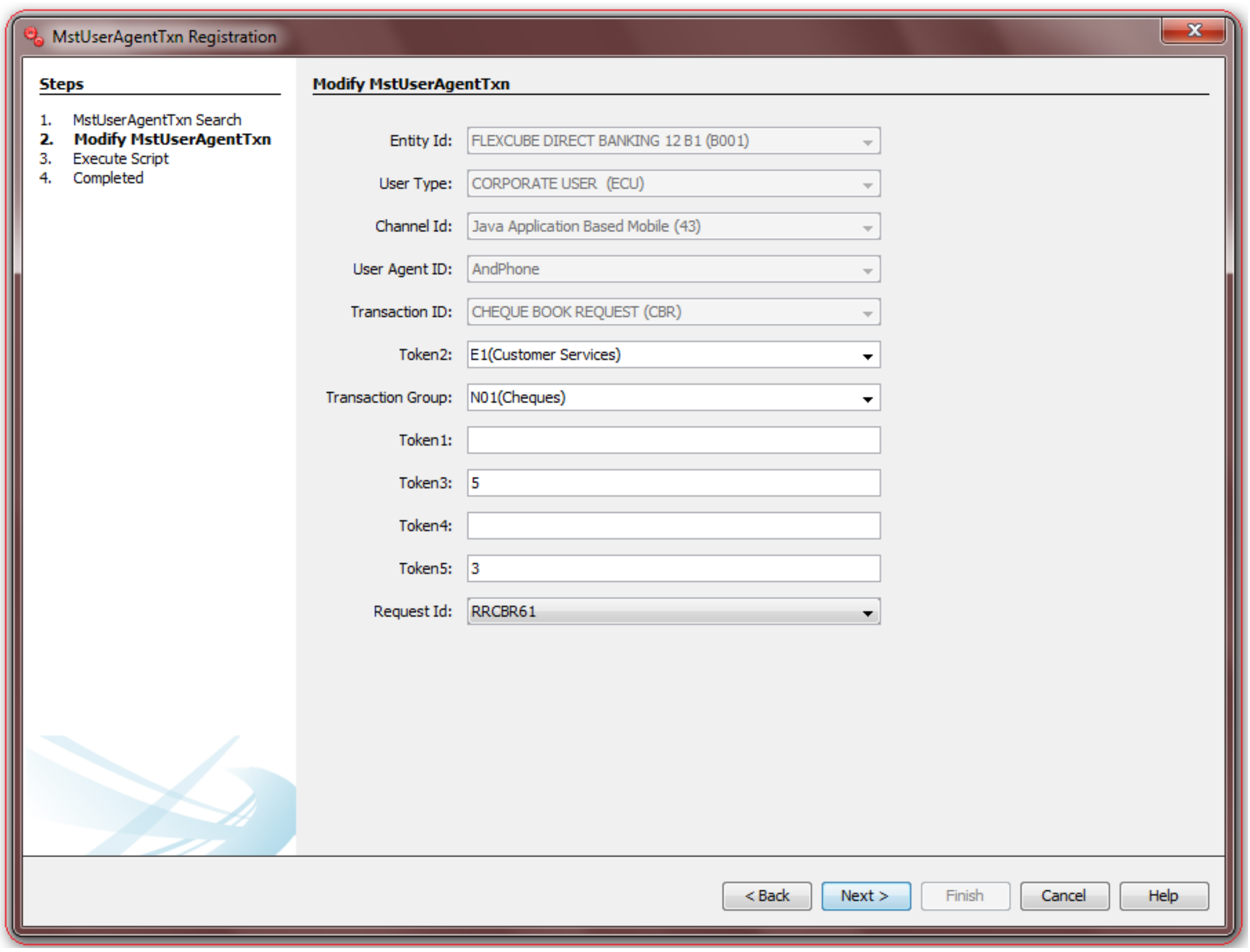
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.

III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstUserAgentTxn - Edit

This step is meant to allow users to input details for a new MstUserAgentTxn configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Entity ID**: Select the Entity ID.
- **User Type**: Select the User Segment.
- **Channel ID**: Select the Channel ID.

- **User Agent ID**: Select the User Agent ID.
- **Transaction ID**: Select the Transaction ID for which the menu needs to be configured.
- **Token2**: Select the main menu under which this transaction will come, description of main menus is defined in appldata table with dataname = 'MENU_TAB_DESC' and datavalue = <Value of this field>. To define a new main menu, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Transaction Group**: Select the group under which this transaction will come, description of group is defined in appldata table with dataname = 'TXN_GROUP_DESC' and datavalue = <Value of this field>. To define a new group, use appldata wizard (Tools → FCDB → ApplData Wizard) to configure the same.
- **Token5**: Enter the Numeric value to define the position of transaction in the transaction group.
- **Request ID**: Select the Registered work flow step id (refer to MstChannelATS table) getting called on click of this transaction from menu. If no value specified first step will be called.

Click **Next** to go to next step.

15. Entity Registration

Oracle FLEXCUBE Direct Banking Application houses the table **“mstentity”** where FCDB Business Entities are registered. Any new Entity to be added in FCDB application needs to be registered here.

MstEntity Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		A unique identifier for the entity. Primary key of table.
DESCRIPTION	VARCHAR2 (100 CHAR)	No		Description of the entity.
TYPENTITY	CHAR (1 CHAR)	No		Entity Type.
ID_PARENT	VARCHAR2 (5 CHAR)	Yes		Parent Id of entity.
IDSORT	NUMBER	No		Sort Id of entity.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open Entity Registration Wizard, Go to Menu Options

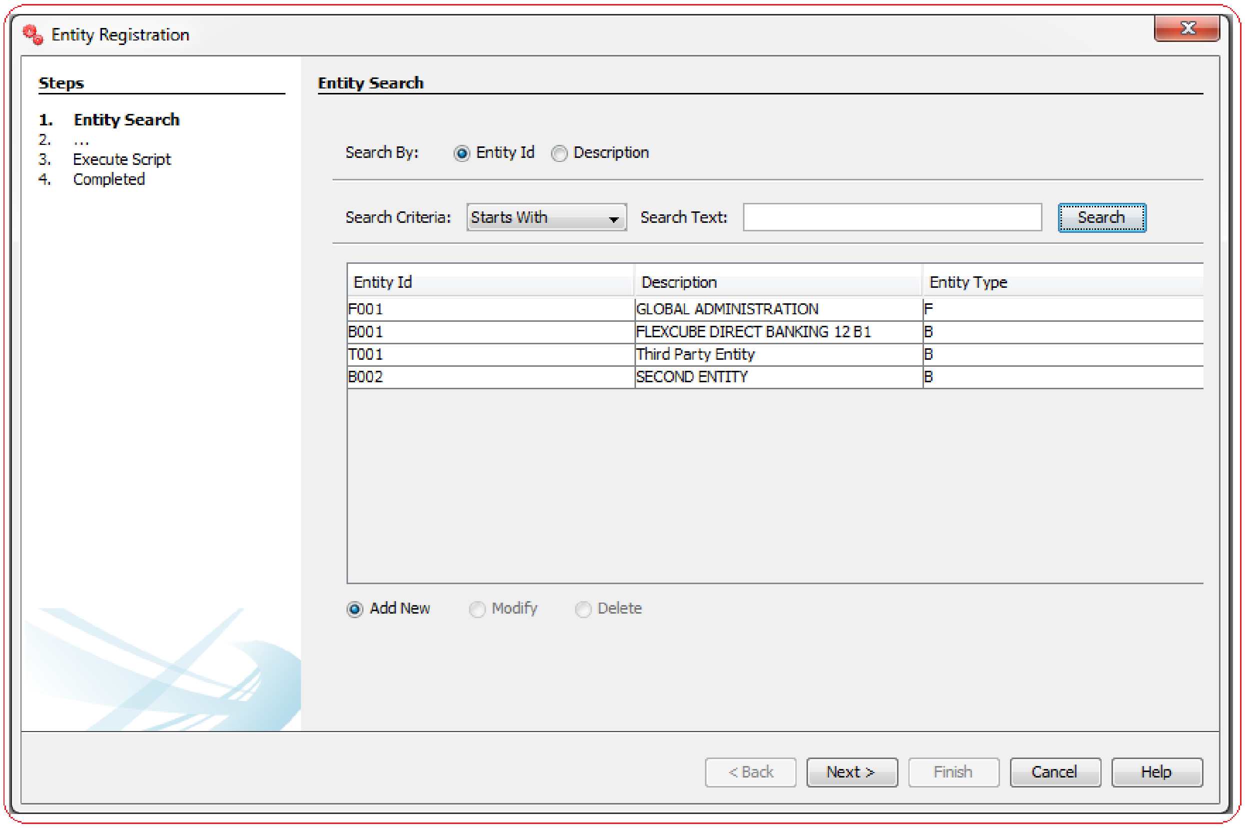
Tools → FCDB → Business Unit Configurations → Entity→ Entity Registration.

Entity - Search

This step is meant to allow the user to search for the existing entries available in **“mstentity”** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.

One can use the "Entity Id" or "Description" parameter to filter the results.

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table **mstentity** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



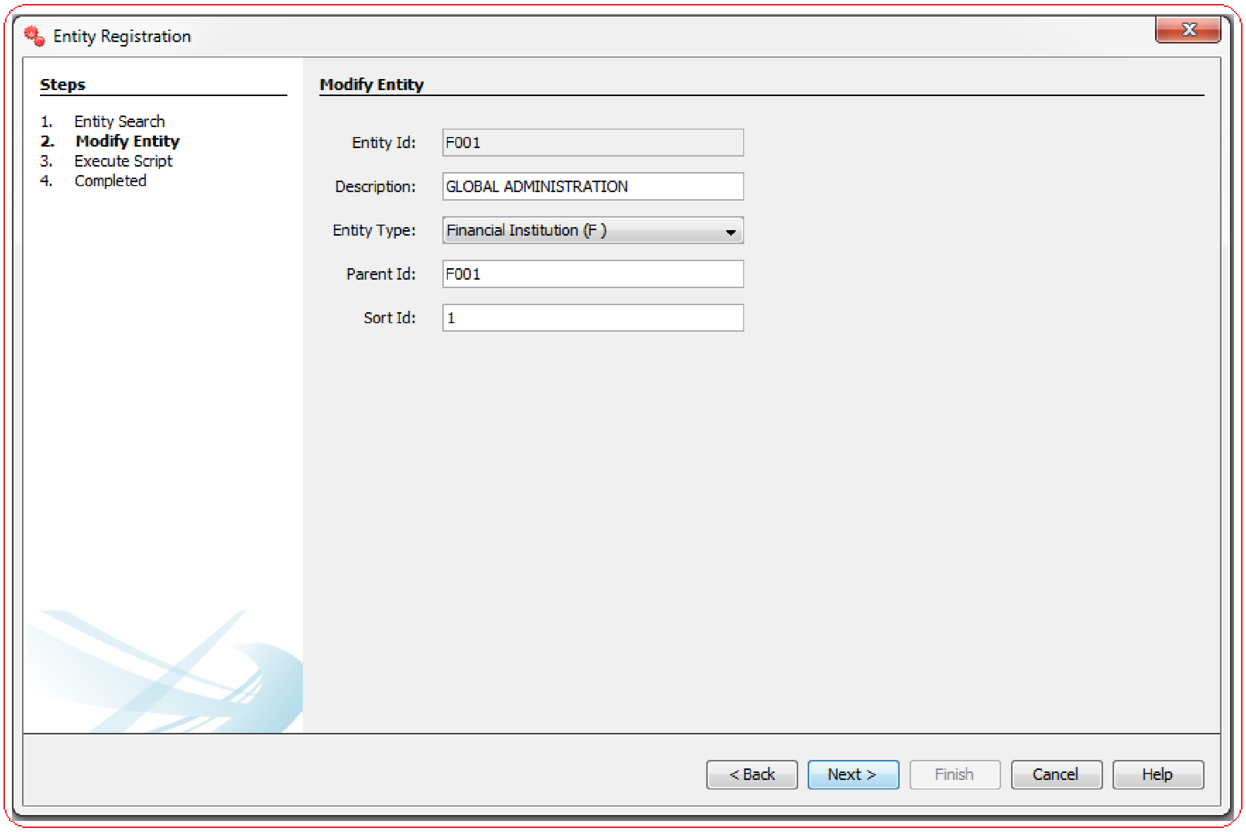
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

<u>Entity - Edit</u>

This step is meant to allow users to input details for a new MstEntity configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Entity ID:** Enter a Unique Id for the Entity.
- **Description:** Enter the Entity Description.
- **Entity Type:** Enter the Entity Type.
- **Parent ID:** Enter the Parent Id of Entity.
- **Sort ID:** Enter a numeric value to identify the position of this Entity relative to other Entities (For e.g. in entity drop downs in FCDB application).

Click **Next** to go to next step.

16. User Type Registration

Oracle FLEXCUBE Direct Banking Application houses the table **mstusertypes** where FCDB Business User-Segments are registered. Any new User Segment to be added in FCDB application needs to be registered here.

MstUserTypes Definition

Column Name	Type	Nullable	Default	Comments
TYPEUSER	VARCHAR2 (3 CHAR)	No		A unique identifier for the usertype. Primary key of table.
DESCRIPTION	VARCHAR2 (100 CHAR)	No		Description of the usertype.
ISENABLED	VARCHAR2 (1 CHAR)	No	'Y'	Flag to enable/disable this User Type in the application.
HASMULTCUST	VARCHAR2 (1 CHAR)	No	'N'	Flag to identify if multiple Customer Mapping is allowed for this user or not.
FLGINTERNALTYPE	VARCHAR2 (1 CHAR)	Yes	'N'	'N' -- Non-Proxy User Type. 'P' -- Proxy User Type.
BASEUSERTYPE	VARCHAR2 (3 CHAR)	No	'ENU'	Base User Type for this User Type.
HASCUSTID	VARCHAR2 (1 CHAR)	No		Flag to identify if Customer ID can be mapped to this user. For e.g. for Admin user this flag can be set to 'N'.
LOOKUPTYPE	VARCHAR2 (1 CHAR)	No		'M' -- Mandate Setup. Set this value if list of accounts to be displayed to the user needs to be filtered on the basis of Mandate Setup done.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open User Type Registration Wizard, Go to Menu Options

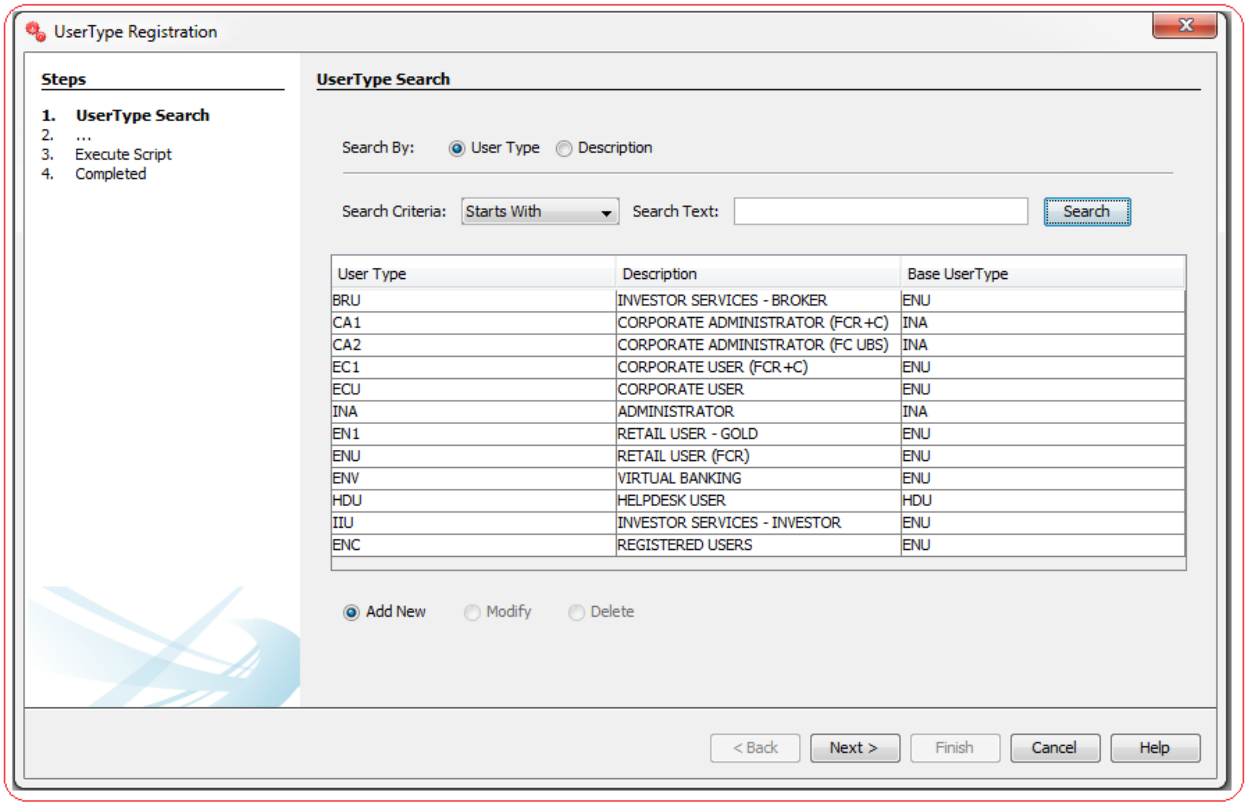
Tools → FCDB → Business Unit Configurations → User Type→ User Type Registration.

UserType - Search

This step is meant to allow the user to search for the existing entries available in **mstusertypes** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.

One can use the "User Type" or "Description" parameter to filter the result as described below.

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table **mstusertypes** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



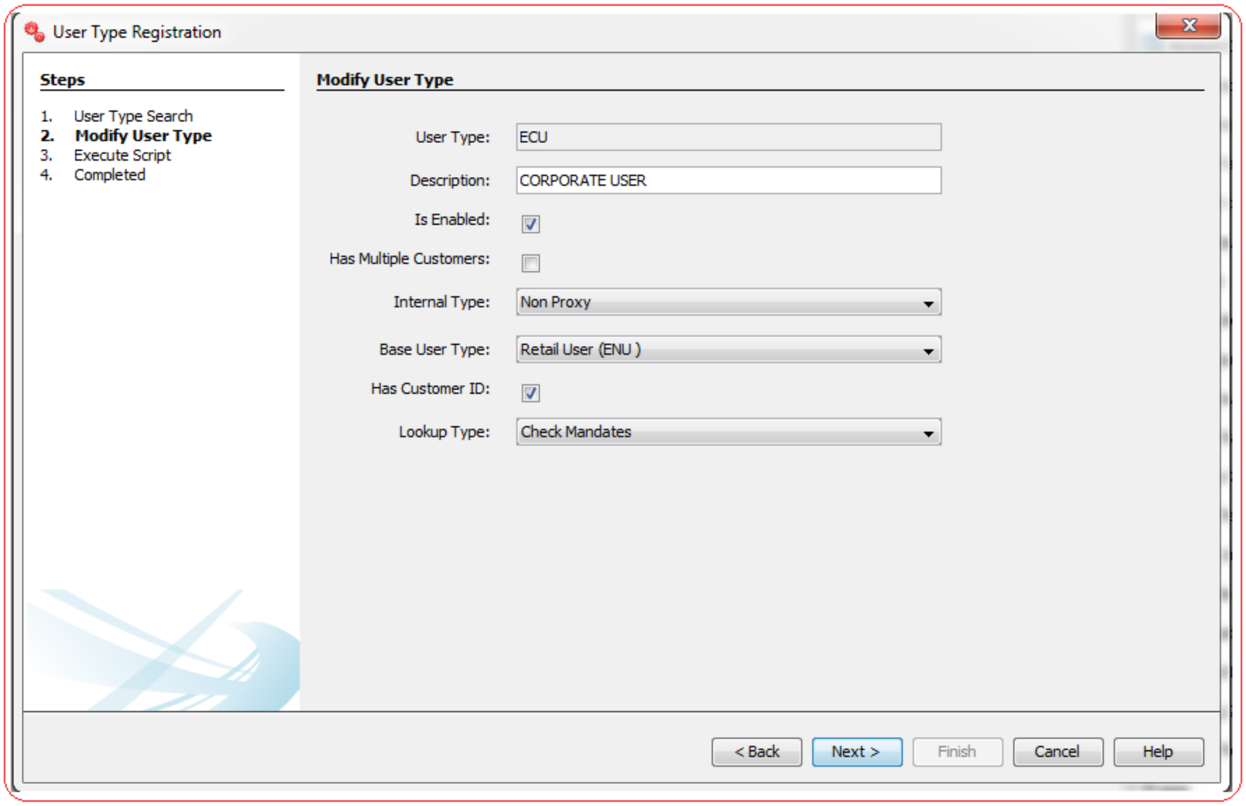
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

<u>UserType - Edit</u>

This step is meant to allow users to input details for a new **MstUserTypes** configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **User Type:** Enter a Unique Id for the User Type.
- **Description:** Enter the User Type Description.
- **Is Enabled:** Flag to enable/disable this user type in the application.
- **Has Multiple Customers:** Flag to identify if multiple Customer Mapping is allowed for this user or not.
- **Internal Type:** Select the type as "Non Proxy" or "Proxy".
- **Base User Type:** Select the Base User Type for this User Type.
- **Has Customer Id:** Flag to identify if Customer ID can be mapped to this user. For e.g. for Admin user this flag can be set to 'N'.
- **Lookup Type:** Select "Check Mandates", if list of accounts to be displayed to the user needs to be filtered on the basis of Mandate Setup done.

Click **Next** to go to next step.

17. Entity – User Type Mapping

Oracle FLEXCUBE Direct Banking Application houses the table **mstentityusertypes** where User Types are mapped to a particular Business Entity. Any new Entity - User Type mapping to be added in FCDB application needs to be registered here.

MstEntityUserTypes Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		Entity ID.
TYPEUSER	VARCHAR2 (3 CHAR)	No		User Type.
DESCRIPTION	VARCHAR2 (100 CHAR)	No		User Type Description.
ISENABLED	CHAR (1 CHAR)	No	Y	Flag to Identify if this mapping is Enabled or Not.
NAMAUTHENTICATOR	VARCHAR2 (255 CHAR)	No	com.iflex.fcat.services.apps.DefaultPasswordAuthenticator	Full Qualified name of the Password Authentication Class implementing the Interface "com.iflex.fcat.services.apps.Authenticator".
LIMITSALLOWED	VARCHAR2 (1 CHAR)	No	N	Flag to identify if transactional limit is allowed for this entity and user type combination.
IDSORT	NUMBER	No	0	Numeric value to identify the position of this usertype in entity-uertype dropdown relative to other usertypes.
REF_USERTYPE	VARCHAR2(255 CHAR)	Yes		Comma separated List of UserTypes which will be available to this UserType when logged in into FCDB Application. This list helps to restrict the visibility of usertypes in Entity-Usertype dropdowns in FCDB Application.
IS_ROLE_CUSTPROF_ALLOWED	CHAR (1 CHAR)	No	N	Flag to display the checkbox "Set As Customer Profile Role" while defining new Role in FCDB Application for this UserType.
TXNINITFLOWHANDLER	VARCHAR2 (255 CHAR)	Yes		Full Qualified name of the Transaction Pin Handler Class implementing the Interface "com.iflex.fcat.services.apps.TransactionInitHandler".
IS_ROLE_DEFAULT_ALLOWED	CHAR (1 CHAR)	No	N	
TYPUSERTOKEN	CHAR (1 CHAR)	No	S	
USERLIFECYCLEEXTHANDLER	VARCHAR2 (255 CHAR)	Yes		Full Qualified name of the User Life Cycle Handler Class

Column Name	Type	Nullable	Default	Comments
				implementing the Interface "com.iflex.fcat.services.apps.User LifeCycleExtendedHandler".
TYPEROLE	CHAR (1 CHAR)	No	B	A (Admin) B (Business) P (Prospect)
PROXYUSERTYPE	VARCHAR2 (3 CHAR)	Yes		Proxy User Type.
LOOKUPTYPE	CHAR (1 CHAR)	Yes		'M' -- Mandate Setup. Set this value if list of accounts to be displayed to the user needs to be filtered on the basis of Mandate Setup done. Value Configured here will override the configuration for LookUpType done while registering new UserType.
HASCUSTOMER	CHAR (1 CHAR)	Yes		P (Primary) L (Linked) A (All) N (None) S (Secondary)
HOSTTYPE	CHAR (1 CHAR)	Yes		Host Model (S: Single host model, M: Multiple host model,N: Not applicable)
ISACTIVEFLAG	CHAR (1 CHAR)	Yes	N	Flag to display Activate User Check Box at the time of User Creation for this UserType.
TOKEN1	NUMBER	Yes		Maximum number of Public Beneficiary Templates allowed for this usertype.
TOKEN2	NUMBER	Yes		Maximum number of Private Beneficiary Templates allowed for this usertype.
TOKEN3	VARCHAR2 (20 CHAR)	Yes	N	Token3.
TOKEN4	VARCHAR2 (20 CHAR)	Yes	N	Flag to make check box for "Alert to Beneficiary" default checked While creating Customer profile for this usertype.
TOKEN5	NUMBER	No	10	Place holder to set value for Maximum number of Cheque book.
TOKEN6	CHAR (1 CHAR)	No	N	Place holder to set the flag for allowing multiple Cheque books.
TOKEN7	VARCHAR2 (20 CHAR)	Yes	Y	Token7.
TOKEN8	VARCHAR2(20 CHAR)	Yes		Place holder to set the flag for allowing Account Nicknames.
TOKEN9	VARCHAR2(20 CHAR)	Yes		Place holder to set the flag for allowing Wealth Management

Column Name	Type	Nullable	Default	Comments
				Customer.
TOKEN10	VARCHAR2(20 CHAR)	Yes		Numeric value to set the priority for the usertype products to be displayed post log-in to prospect usertype (Origination Module).
TOTALSET	NUMBER(2, 0)	Yes	0	Total Security Question Set.
QUESPERSET	NUMBER(2, 0)	Yes	0	Total Questions per Set.
SECLOCKCOUNT	NUMBER(2, 0)	Yes	0	Maximum Number of attempts allowed for Security Questions.
NBRSECQUESASK	NUMBER(2, 0)	Yes	0	Number of Security Questions to be asked for Authentication.
ISCASESENSITIVE	CHAR(1 CHAR)	Yes	N	Flag to set if Security Answers are Case-Sensitive or Not.

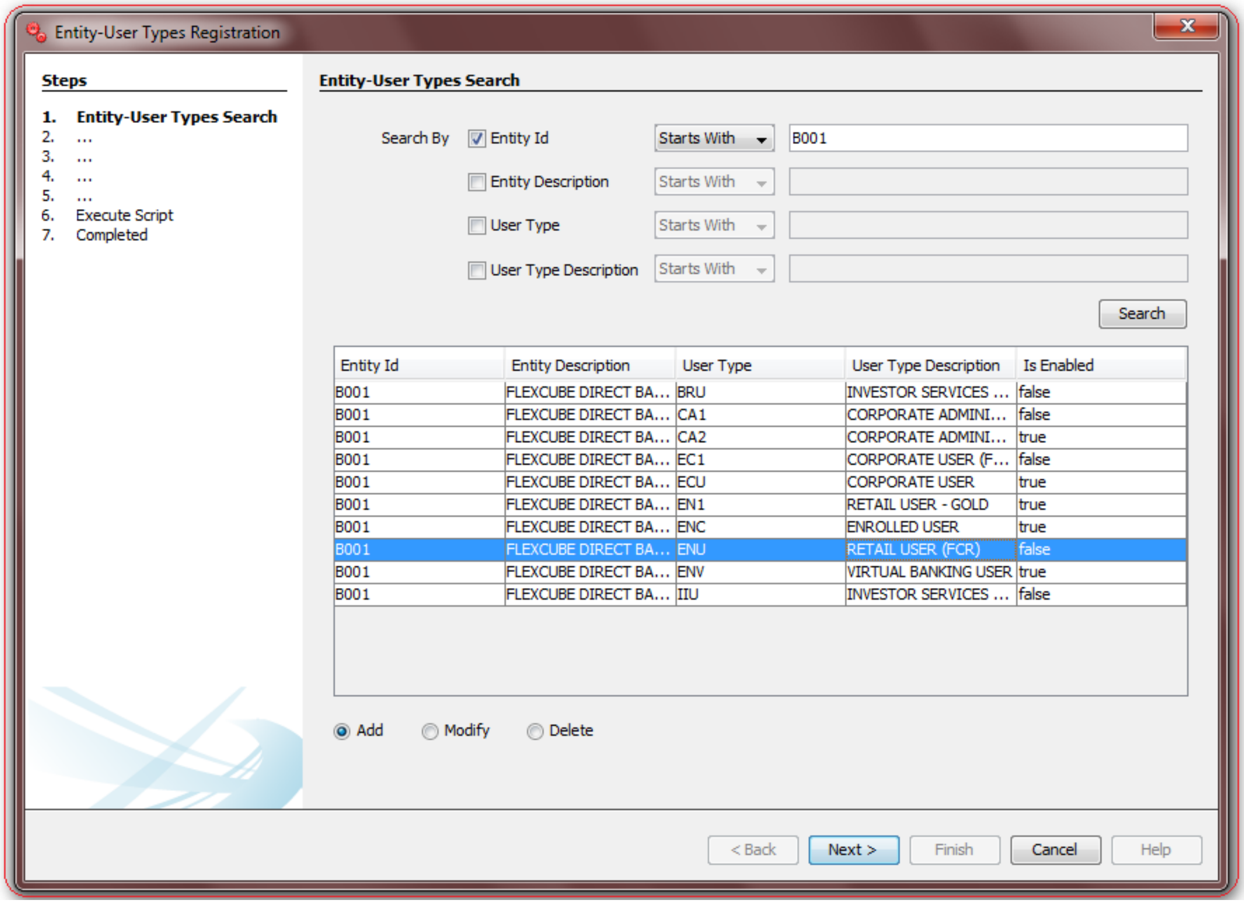
A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open Entity - User Type Mapping Wizard, Go to Menu Options

Tools → FCDB → Business Unit Configuration → User Type → Entity - User Type Mapping

Entity - User Type Mapping - Search

This step is meant to allow the user to search for the existing entries available in MstEntityUserTypes table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.
One can use the "Entity ID" or "Entity Description" or "User Type" or "User Type Description" parameter to filter the result as described below,
Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available entries in the table MstEntityUserTypes matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



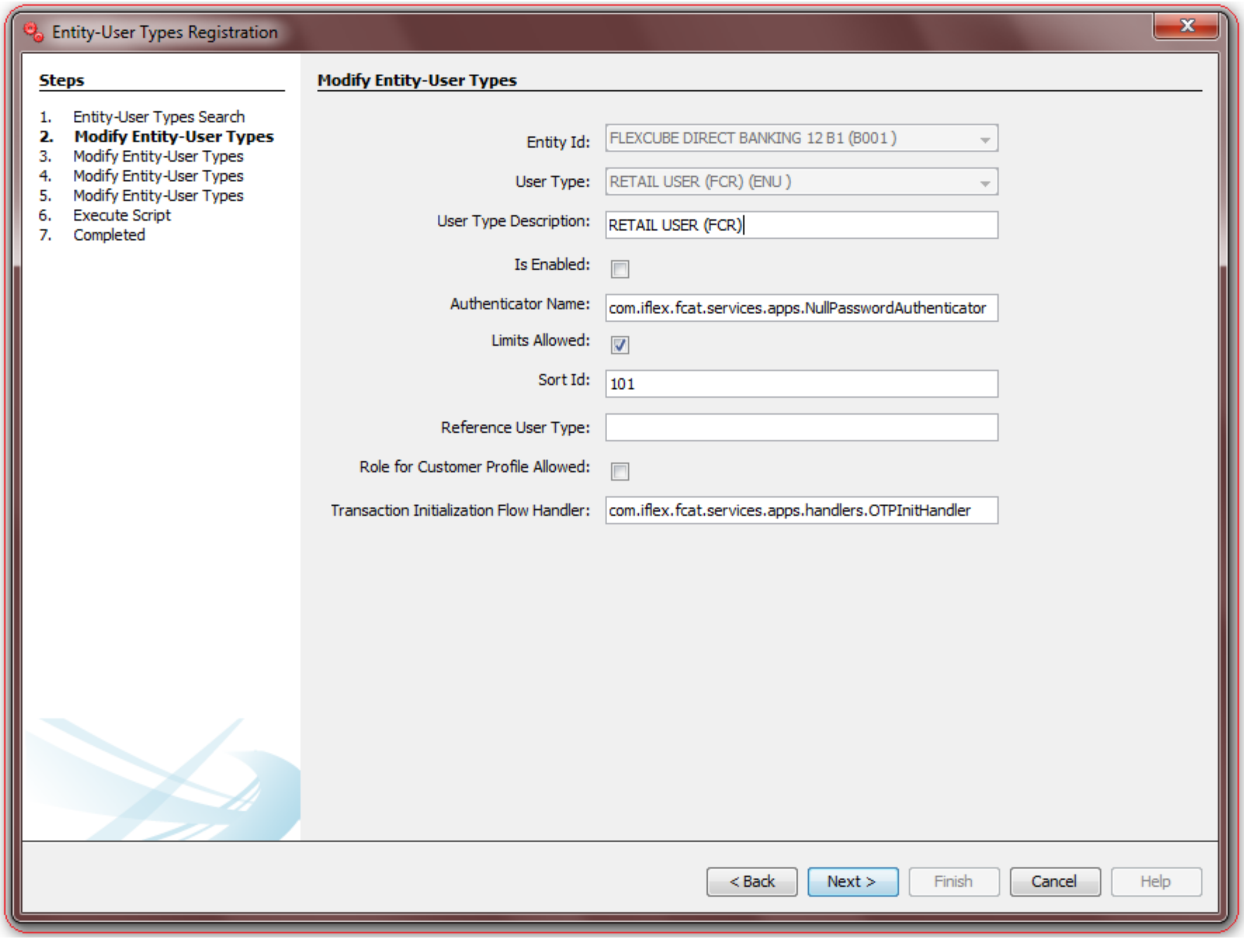
Depending upon the search output one can do the following operations

I. **Add New:** Select this radio button to add new property in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Entity - User Type Mapping - Edit

This step is meant to allow users to input details for a new MstEntityUserTypes configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



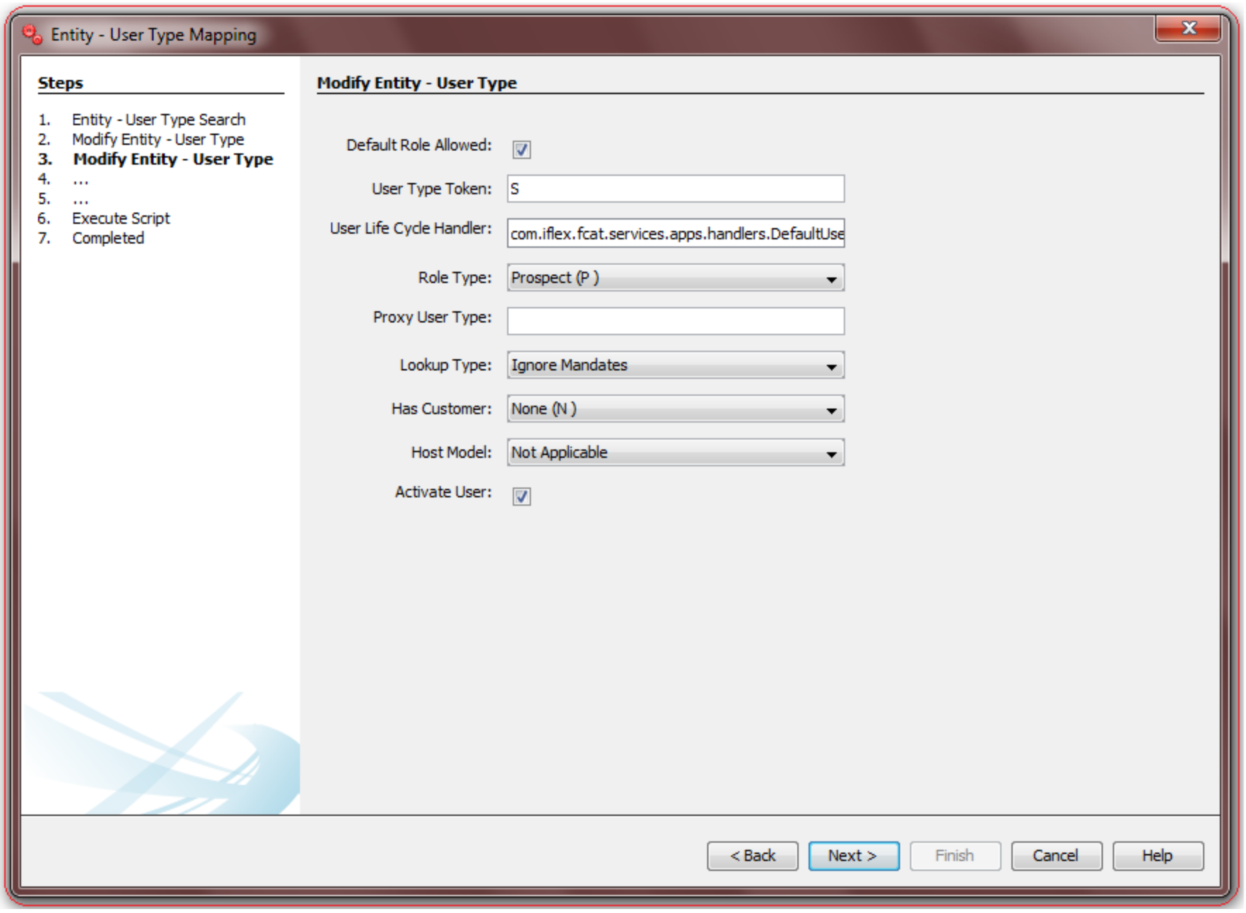
List of the User Input fields and their description are as follows:

- **Entity ID**: Select the Entity ID.
- **User Type**: Select the User Type.
- **User Type Description**: Enter the User Type Description.
- **Is Enabled**: Un-check this check box if mapping of Entity-UserType needs to be disabled.
- **Authenticator Name**: Enter the Password Authentication Class. A Password Authenticator is a concrete implementation of "com.iflex.fcat.services.apps.Authenticator". User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes implementing the Interface.
- **Limits Allowed**: Flag to identify if transactional limit is allowed for this entity and user type combination.
- **Sort ID**: Enter Numeric value to identify the position of this usertype in entity-uertype dropdown relative to other usertypes.

- **Reference User Type**: Enter Comma separated List of UserTypes which will be available to this UserType when logged in into FCDB Application. This list helps to restrict the visibility of usertypes in Entity-Usertype dropdowns in FCDB Application.

- **Role for Customer Profile Allowed**: Flag to display the checkbox "Set As Customer Profile Role" while defining new Role in FCDB Application for this UserType.

- **Transaction Initialization Flow Handler** : Enter Full Qualified name of the Transaction Pin Handler Class implementing the Interface "com.iflex.fcat.services.apps.TransactionInitHandler".User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes implementing the Interface.



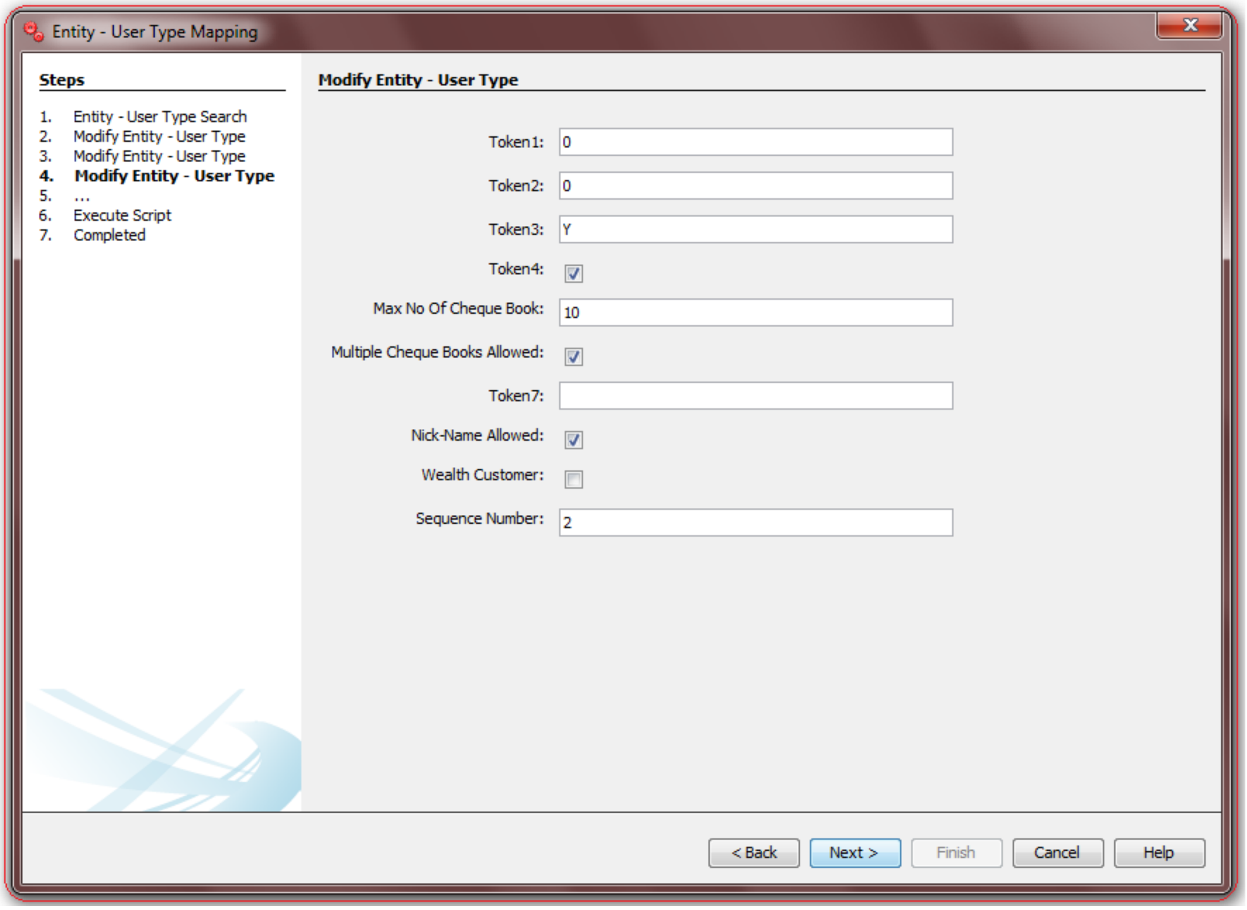
- **Default Role Allowed**: Ignore this field.

- **User Type Token**: Ignore this field.

- **UserLifeCycleExtendedHandler** : Enter Full Qualified name of the Transaction Pin Handler Class implementing the Interface "com.iflex.fcat.services.apps.UserLifeCycleExtendedHandler".User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes implementing the Interface.

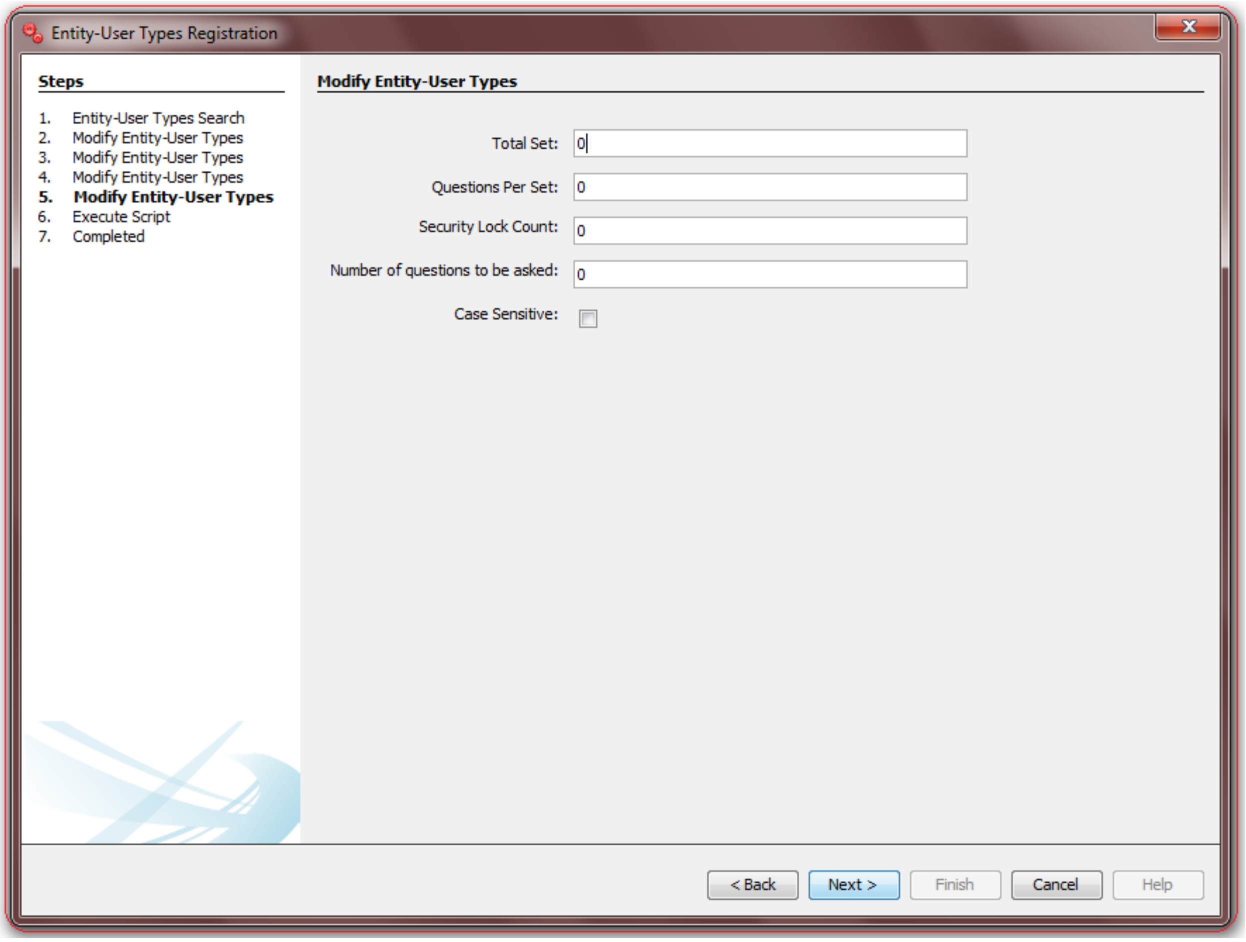
- **Role Type**: Select the Role Type.

- **Proxy User Type**: Enter the Proxy User Type.
- **LookUp Type**: Select the LookUp Type.
- **Has Customer**: Select the Has Customer.
- **Host Model**: Ignore this field.
- **Activate User**: Check this Flag to display Activate User Check Box at the time of User Creation for this UserType.



- **Token1**: Maximum number of Public Beneficiary Templates allowed for this usertype.
- **Token2**: Maximum number of Private Beneficiary Templates allowed for this usertype.
- **Token3**: Ignore this field.
- **Token4**: Flag to make check box for "Alert to Beneficiary" default checked While creating Customer profile for this usertype.
- **Max No Of Cheque Book**: Enter the Max No Of Cheque Book.
- **Multiple Cheque Books Allowed**: Select the Multiple Cheque Books Allowed.

- **Token7**: Ignore this field.
- **Nick-Name Allowed**: Check this flag to allow Account Nicknames.
- **Wealth Customer**: Check this flag to allow Wealth Management Customer.
- **Sequence Number**: Enter a Numeric value to set the priority for the usertype products to be displayed post log-in to prospect usertype (Origination Module).



- **Total Set**: Enter the Total Security Question Set.
- **Questions Per Set**: Enter the number of Questions Per Set.
- **Security Lock Count**: Enter the Maximum Number of attempts allowed for Security Questions.
- **Number of questions to be asked**: Enter the Number of Security Questions to be asked for Authentication.
- **Case Sensitive**: Flag to set if Security Answers are Case-Sensitive or Not.

Click **Next** to go to next step.

18. Channel Registration

Oracle FLEXCUBE Direct Banking Application houses the table **mstdevice** where FCDB Application Channels are registered. Any new Channel to be added in FCDB application needs to be registered here.

MstDevice Definition

Column Name	Type	Nullable	Default	Comments
IDDEVICE	CHAR(2 CHAR)	No		A unique identifier for the device id. Primary key of table.
NAMDEVICE	VARCHAR2 (30 CHAR)	No		Description of the device.
CONTENTTYPE	VARCHAR2 (50 CHAR)	Yes		Content Type.
IDCHANNEL	CHAR(3 CHAR)	No		Channel Id.Unique key of table.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

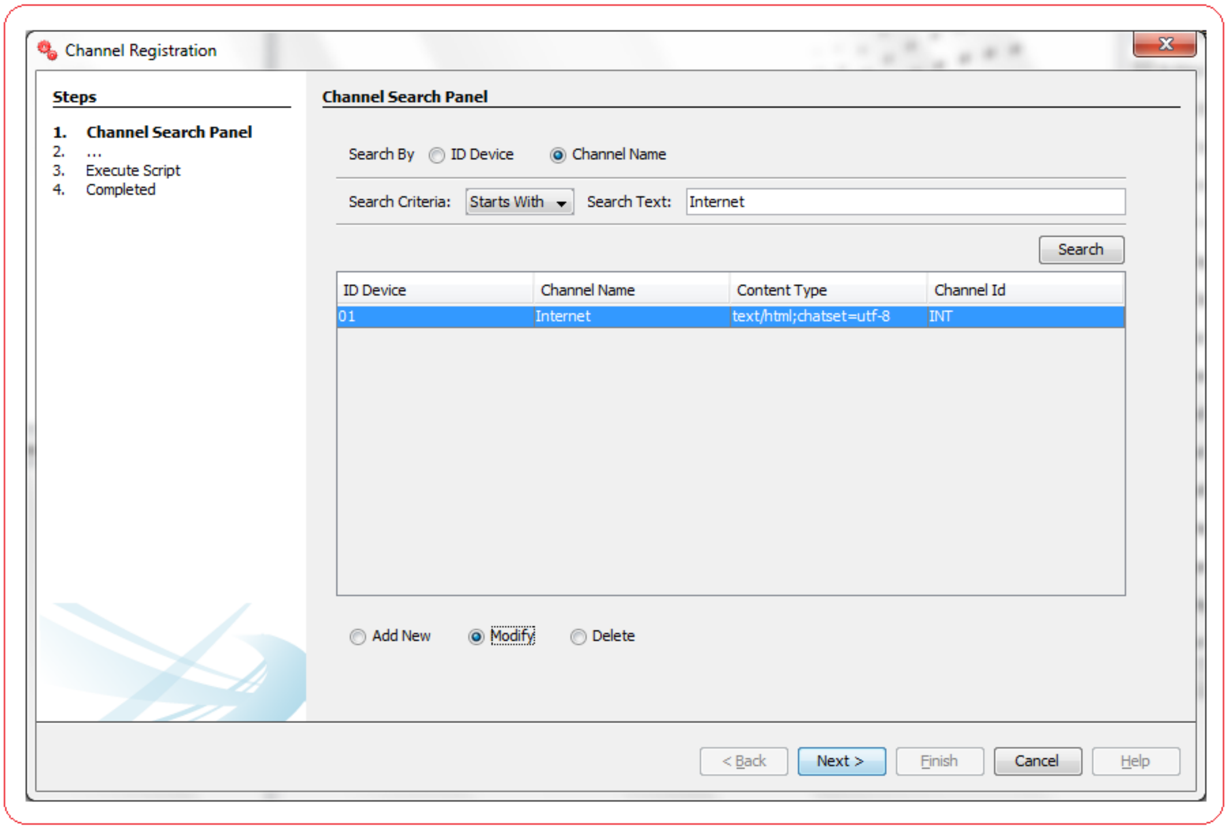
To Open Channel Registration Wizard, Go to Menu Options

Tools → FCDB → Business Unit Configurations → Channel→ Channel Registration.

Channel – Search

This step is meant to allow the user to search for the existing entries available in **mstdevice** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.

One can use the "ID Device" and "Device Name" parameter to filter the result as described below. Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table **mstdevice** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



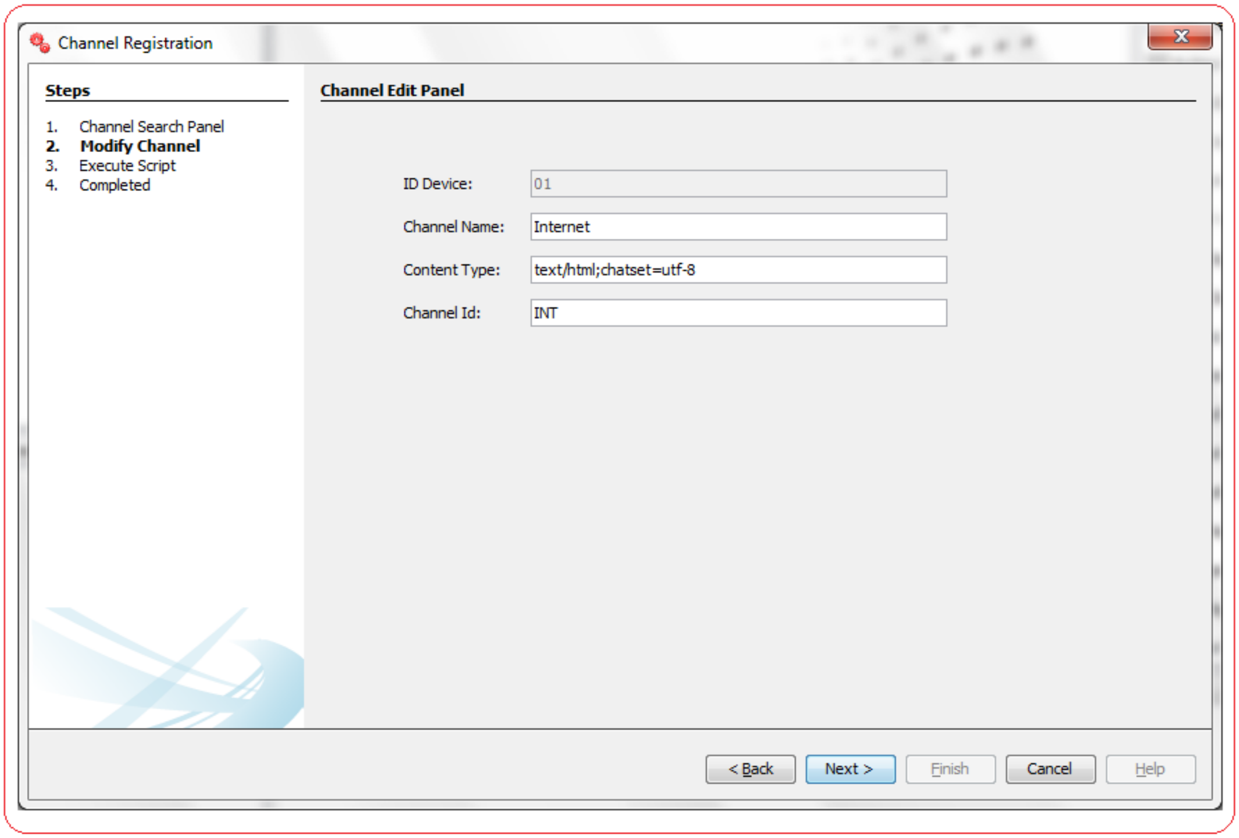
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Channel – Edit

This step is meant to allow users to input details for a new MstDevice configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Device Id:** Enter a two letter numeric value for the channel.
- **Name Device:** Enter the channel description.
- **Content Type:** Enter the Content Type.
- **Channel ID :** Enter a three character identifier for this channel.

Click **Next** to go to next step.

19. Entity - User Type - Channel Grouping

Oracle FLEXCUBE Direct Banking Application houses the table **mstentityuserchannelgrp** where bank can define Channel Groups. Channel group's helps bank to form a group such that a single set of security credentials can be maintained across all channels.

This eliminates the need to maintain multiple user id's and passwords to login and transact on the application from various available channels. The channel grouping is made available only for following channels:

1. Internet Banking
2. Browser Based Mobile Banking
3. Application Based Mobile Banking

SMS Banking would always continue to be considered as a separate group having just the SMS banking channel.

The bank can configure the groups such that any one or more of the above three channels can form a group. However, it is mandatory that no two groups would have the same channel in them for the same entity-user type combination.

MstEntityUserChannelGrp Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		Entity ID.
TYPEUSER	VARCHAR2 (3 CHAR)	No		User Type.
IDGROUP	VARCHAR2 (5 CHAR)	No		Group ID for grouping multiple channels.
POLICYIDPASSWORD	VARCHAR2 (50 CHAR)	No		The Login Password Policy applicable to the Channel Group.
POLICYIDTRANSACTION	VARCHAR2 (50 CHAR)	Yes		The Transaction Password Policy applicable to the Channel Group.
POLICYIDUSER	VARCHAR2 (50 CHAR)	No		The User ID Policy applicable to the Channel Group.
POLICYCODEACTIVATION	VARCHAR2 (50 CHAR)	Yes		The Activation Code Policy applicable to the Channel Group.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

To Open Entity – User Type - Channel Grouping Wizard, Go to Menu Options

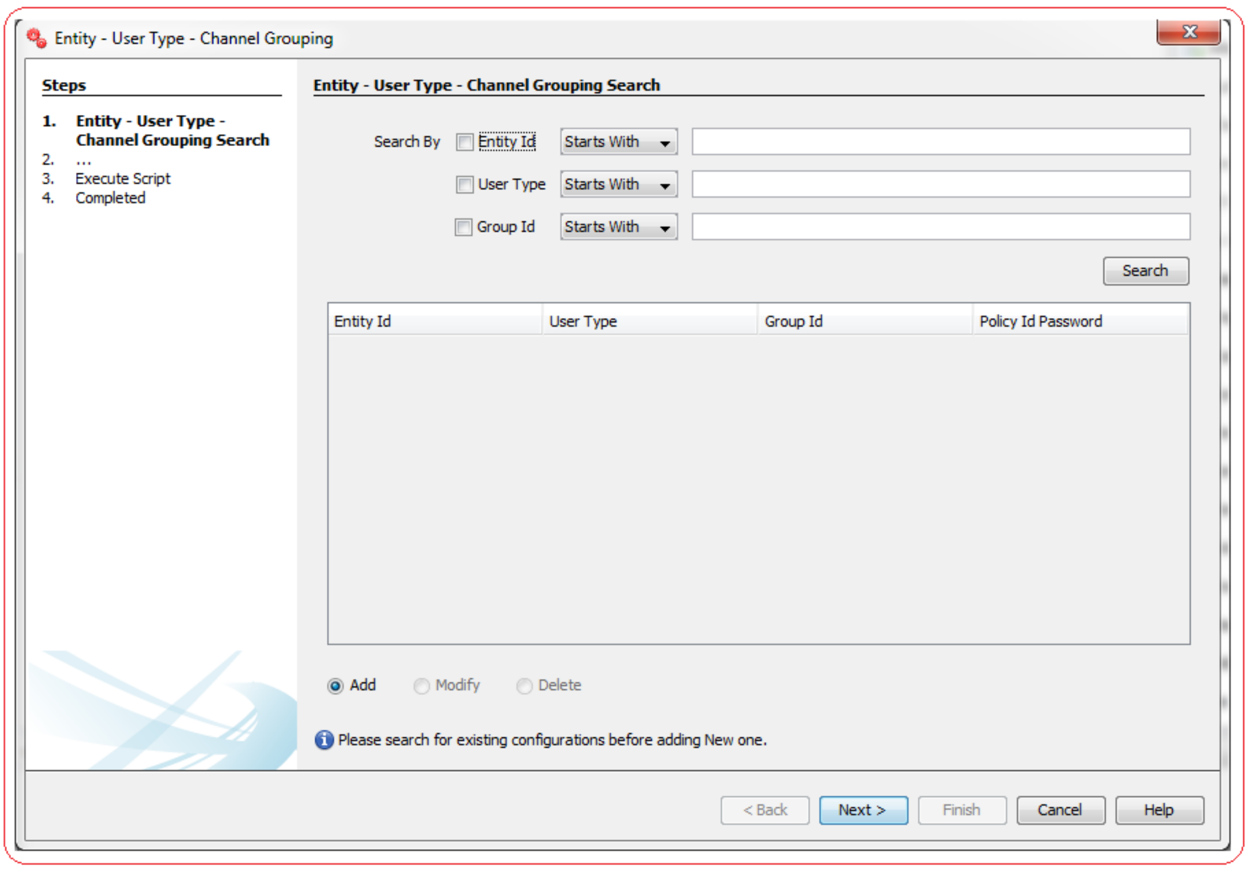
Tools → FCDB → Business Unit Configurations → Channel→ Entity – User Type - Channel Grouping.

Entity - User Type - Channel Grouping - Search

This step is meant to allow the user to search for the existing entries available in **mstentityuserchannelgrp** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.

One can use the "Entity ID" or "User Type" or "Group ID" parameter to filter the result as described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table mstentityuserchannelgrp matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



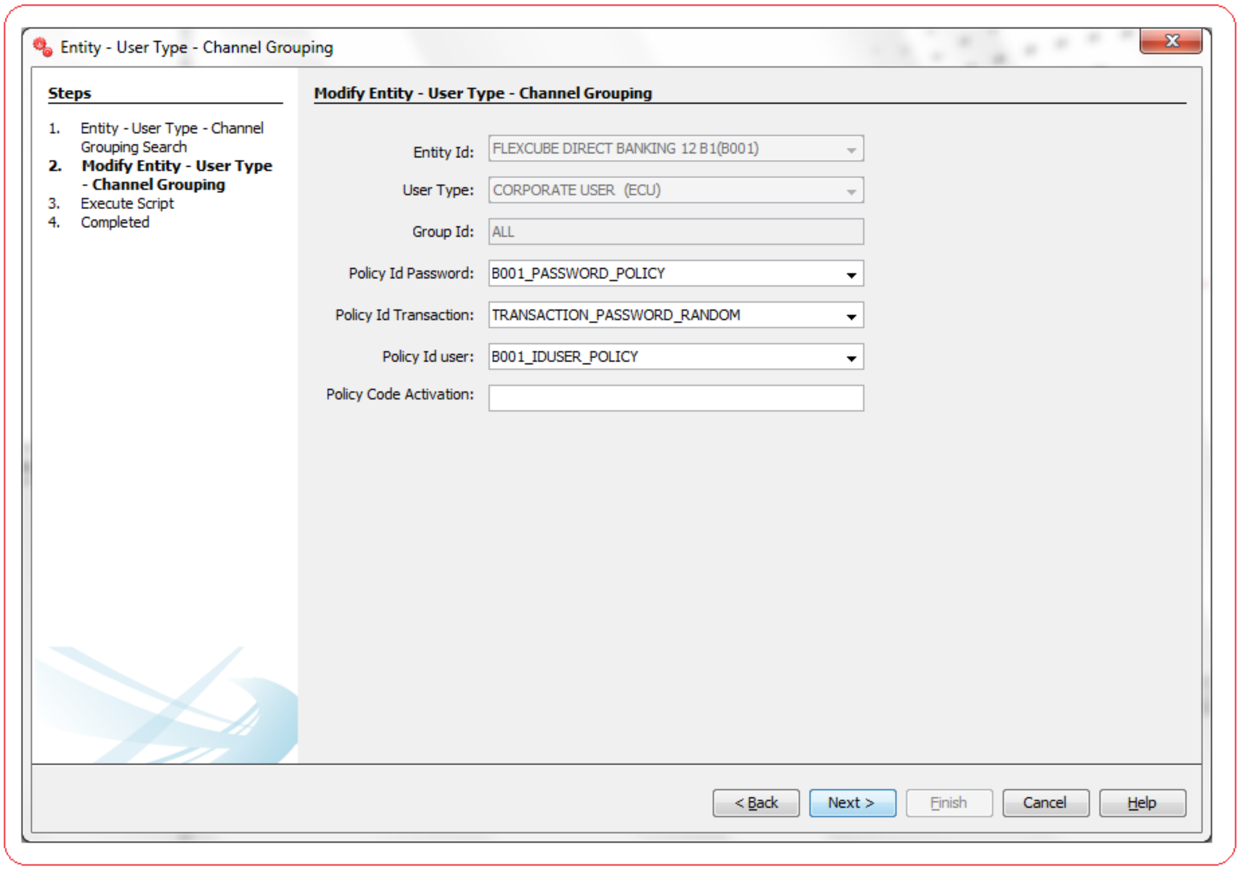
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Entity - User Type - Channel Grouping - Edit

This step is meant to allow users to input details for a new MstentityUserChannelGrp configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Entity ID**: Select the Entity ID.
- **User Type**: Select the User Type.
- **Group Id**: Enter the Group Id.
- **Policy Id Password**: Select or Enter the Login Password Policy applicable to the Channel Group.
- **Policy Id Transaction**: Select or Enter the Transaction Password Policy applicable to the Channel Group.
- **Policy Id User**: Select or Enter the User ID Policy applicable to the Channel Group.
- **Policy Code Activation**: Enter the Activation Code Policy applicable to the Channel Group.

Click **Next** to go to next step.

20. Entity - User Type - Channel Mapping

Oracle FLEXCUBE Direct Banking Application houses the table **mstentityuserchannel** where a channel is mapped to particular Entity, UserType and Group defined in **mstentityuserchannelgrp**.

MstEntityUserChannel Definition

Column Name	Type	Nullable	Default	Comments
ID_ENTITY	VARCHAR2 (5 CHAR)	No		Entity ID.
TYPEUSER	VARCHAR2 (3 CHAR)	No		User Type.
IDCHANNEL	VARCHAR2 (2 CHAR)	No		Channel ID which needs to be mapped.
POLICYIDPASSWORD	VARCHAR2 (50 CHAR)	No		The Login Password Policy applicable to the Channel Id.
POLICYIDTRANSACTION	VARCHAR2 (50 CHAR)	Yes		The Transaction Password Policy applicable to the Channel Id.
POLICYIDUSER	VARCHAR2 (50 CHAR)	No		The User ID Policy applicable to the Channel Id.
POLICYCODEACTIVATION	VARCHAR2 (50 CHAR)	Yes		The Activation Code Policy applicable to the Channel Id.
EXTENDEDRESPONSEHANDLER	VARCHAR2 (255 CHAR)	Yes	‘com.iflex.fcat.services.apps.DefaultExtendedResponseHandler’	Full Qualified name of the Service Response Handler class extending the class "com.iflex.fcat.services.ExtResponseHandler".
REFPATHUI	VARCHAR2 (255 CHAR)	Yes		Relative paths to datafiles UI folder. # separated values to determine path to search GUI XSL's.
SESSIONTIMEOUT	NUMBER	No	900	Time in milli seconds to make idle session inactive.
MENUTYPE	VARCHAR2 (2 CHAR)	No	‘L’	Day 0 menu type for Entity-UserType-Channel combination. For Contemporary D - Dashboard - Task bar based DT - Dashboard - Top based DL - Dashboard - Left based For Classic L - Left menu based T - Top menu based TL - Group menu based
LOGINLAYOUT	CHAR(1 CHAR)	No	‘L’	Day 0 layout for Entity-UserType-Channel combination. Classic (L), Contemporary(C) or Simple(S).

Column Name	Type	Nullable	Default	Comments
ACTIVETIMEOUT	NUMBER	No	1200	Time in milli seconds to make session inactive.
IDGROUP	VARCHAR2 (5 CHAR)	Yes		Group ID defined in mstentityuserchannelgrp.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

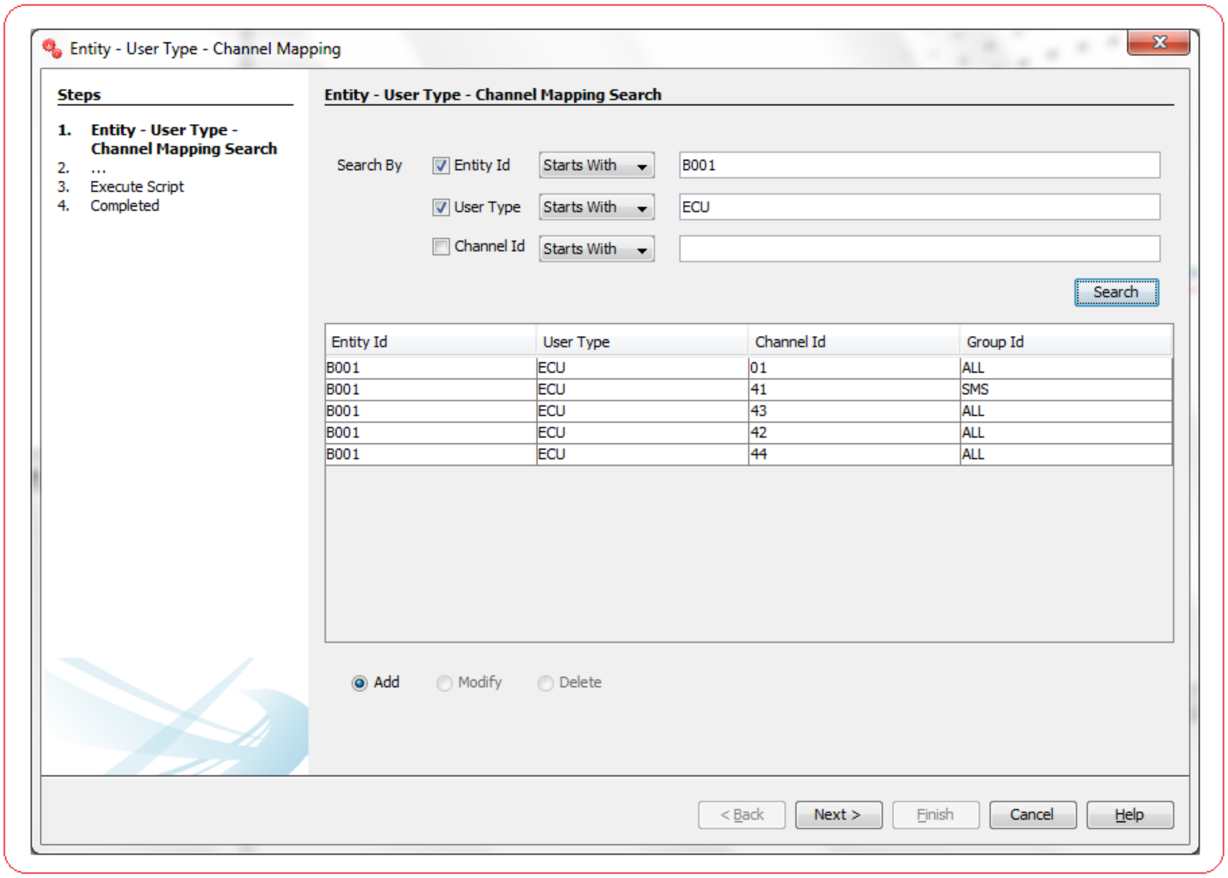
To Open Entity – User Type - Channel Grouping Wizard, Go to Menu Options

Tools → FCDB → Business Unit Configurations → Channel → Entity – User Type - Channel Mapping.

Entity - User Type - Channel Mapping - Search

This step is meant to allow the user to search for the existing entries available in **MstEntityUserChannel** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.
One can use the "Entity ID" or "User Type" or "Channel Id" parameter to filter the result as described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table **MstEntityUserChannel** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



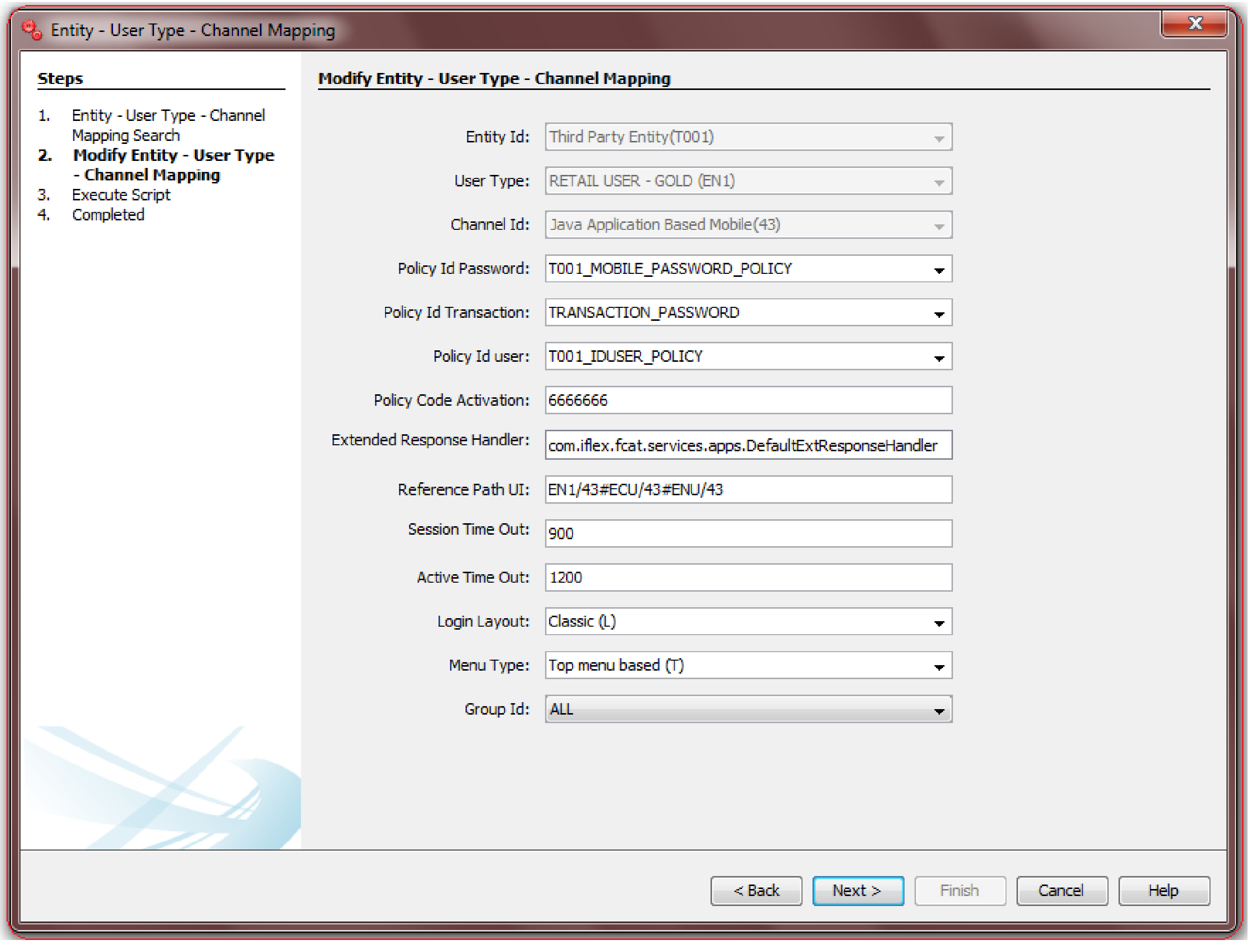
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Entity - User Type - Channel Mapping - Edit

This step is meant to allow users to input details for a new MstEntityUserChannel configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Entity ID**: Select the Entity ID.
- **User Type**: Select the User Type.
- **Channel Id**: Select the Channel which needs to be mapped.
- **Policy Id Password**: Select or Enter the Login Password Policy applicable to the Channel Id.
- **Policy Id Transaction**: Select or Enter the Transaction Password Policy applicable to the Channel Id.
- **Policy Id User**: Select or Enter the User ID Policy applicable to the Channel Id.
- **Policy Code Activation**: Enter the Activation Code Policy applicable to the Channel Id.
- **Extended Response Handler**: Enter a valid Full Qualified name of the Response Handler class extending the class **"com.iflex.fcat.services.ExtResponseHandler"**. User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes extending ExtResponseHandler Class.
- **Reference Path UI**: Enter the Relative paths to datafiles UI folder. # separated values to determine path to search GUI XSL's.
- **Session Time Out**: Enter the Time in milli seconds to make idle session inactive.
- **Active Time Out**: Enter the Time in milli seconds to make session inactive.

- **Login Layout**: Select or Enter the Day 0 layout for Entity-UserType-Channel combination. Classic (L), Contemporary(C) or Simple(S).
- **Menu Type**: Select or Enter the Day 0 menu type for Entity-UserType-Channel combination. For Contemporary
  - D - Dashboard - Task bar based
  - DT - Dashboard - Top based
  - DL - Dashboard - Left based

  For Classic
  - L - Left menu based
  - T - Top menu based
  - TL - Group menu based
- **Group Id**: Select the Group id Value.

Click **Next** to go to next step.

21. MstFormats

Oracle FLEXCUBE Direct Banking Application houses the table **mstformats** where various application format configurations are added for entities like Date, Currency and Numbers.

MstFormats Definition

Column Name	Type	Nullable	Default	Comments
NAMEFORMAT	VARCHAR2 (255 CHAR)	No		Name of the format.
TYPEFORMAT	CHAR2(1 CHAR)	No		D (Date) C (Currency) X (Special)
IDLANG	VARCHAR2 (3 CHAR)	No		Language Id.
TYPEPATTERN	VARCHAR2 (255 CHAR)	No		Output Data Pattern. Value in the field applies the localized pattern on “Output Data”.
NAMLOCALE	VARCHAR2 (5 CHAR)	No		Locale Name. Value in the field applies the format symbols on the “Output Data”.
TYPEINPPATTERN	VARCHAR2 (255 CHAR)	Yes		Input Data Pattern. Input Data Pattern must match with Targeted/Provided Data value Pattern for successful parsing in java class.
CLASSNAME	VARCHAR2 (255 CHAR)	Yes		Full Qualified name of the formatter class extending the class “com.iflex.fcat.infra.JFFormatter”.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

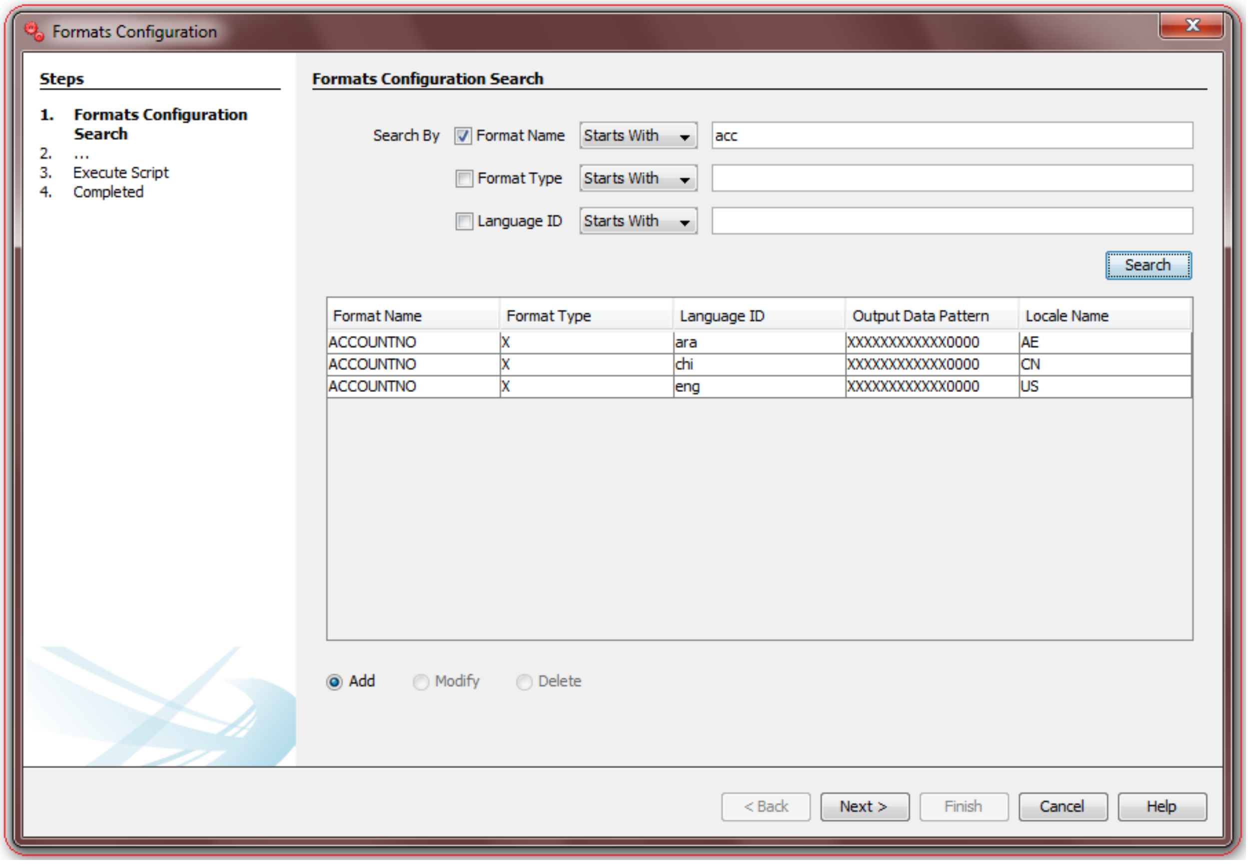
To Open Formats Configuration Wizard, Go to Menu Options

Tools → FCDB → Others → Formats Configuration.

Formats Configuration - Search

This step is meant to allow the user to search for the existing entries available in **mstformats** table of the connected Database. It is always advisable to search for existing entries before adding a new property in this table.
One can use the "Format Name" or "Format Type" or "Language ID" parameter to filter the result described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available properties in the table **mstformats** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



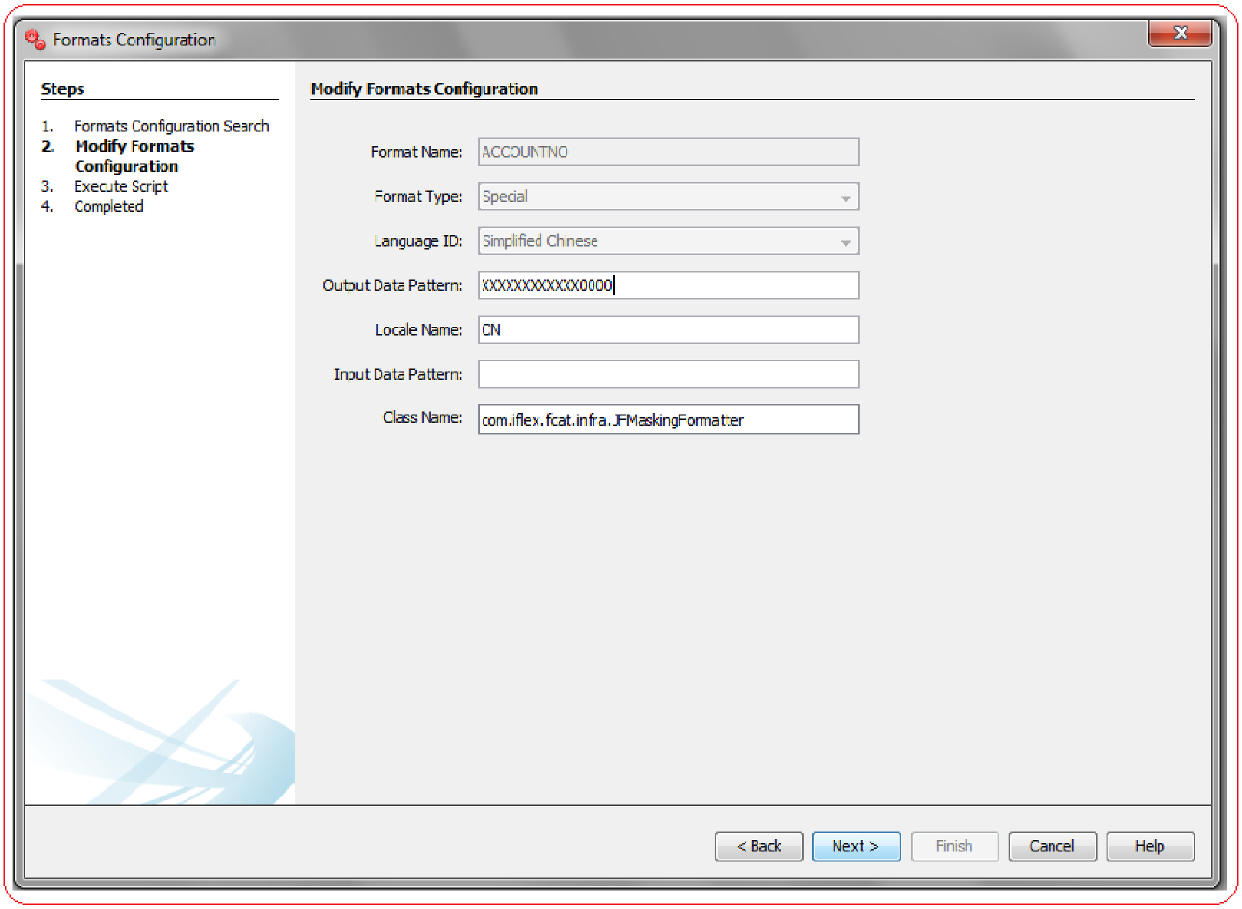
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

Formats Configuration - Edit

This step is meant to allow users to input details for a new MstFormats property, modify details of an existing one or view details of an existing one if user wants to delete the same.



List of the User Input fields and their description are as follows:

- **Format Name** : Enter the Format Name.
- **Format Type** : Select the Format Type.
- **Language Id** : Select the Language Id.
- **Output Data Pattern** : Enter the Output Data Pattern. According to Output Data Pattern, Output Data will get formatted.
- **Locale Name** : Enter the Locale Name. Based on Locale Name, Format Symbols will get apply on result Output Data.
- **Input Data Pattern** : Enter the Input Data Pattern. Input Data Pattern must match with Targeted/Provided DataValue Pattern for successful parsing in java class.
- **Class Name** : Enter a valid Full Qualified name of the Formatter class extending the class "com.iflex.fcat.infra.JFFormatter". User can use auto-suggest on this field by pressing Ctrl+Space and selecting from the list of available Classes extending JFFormatter Class.

Click **Next** to go to next step.

22. MstMobData

Oracle FLEXCUBE Direct Banking Application houses the table **mstmobdata** where various mobile functionalities with respect to GUI and transaction are added which has a significant impact on the behavior of the application.
e.g. - This table is used to invoke many behaviors ,few of them are listed below:

1. Show/Hide the GUI components.
2. QR code operation: This operation is to perform QR payments where the transferor and transferee use QR codes to perform transactions.
3. NFC Payment: This operation is to perform NFC payments where the payment happens over NFC.
4. Future Date: This operation is to set future dates of payments.
5. Currency Format: This operation is to set currency formats.

MstMobData Definition

Column Name	Type	Nullable	Default	Comments
DATANAME	VARCHAR2(50 CHAR)	No		A unique identifier for the mstmobdata. Primary key of table.
DATACODE	VARCHAR2 (100 CHAR)	No		A unique identifier for the mstmobdata. Primary key of table
USERAGENT	VARCHAR2 (100 CHAR)	No		A unique identifier for the mstmobdata. Primary key of table
VALUESTRING	VARCHAR2(4000 CHAR)	Yes		Value String.

A step wise work flow has been provided in this tool to allow user to Add/Modify/Delete these properties.

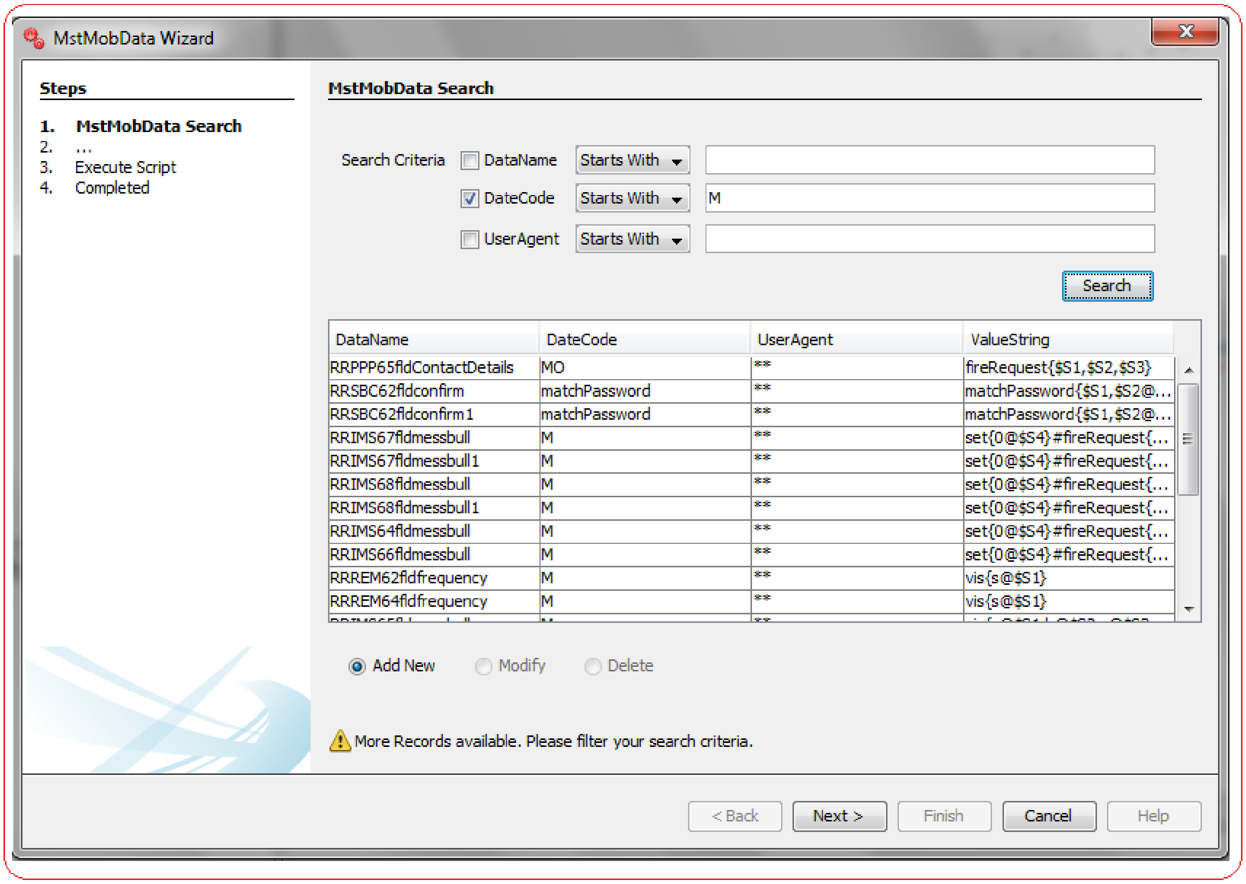
To Open MstMobData Wizard, Go to Menu Options

Tools → FCDB → Others → MstMobData Registration.

MstMobData Registration – Search

This step is meant to allow the user to search for the existing entries available in **mstmobdata** table of the connected Database. It is always advisable to search for existing entries before adding a new configuration in this table.
One can use the "Dataname" and "Datacode" and "UserAgent" parameter to filter the result as described below,

Select corresponding search criteria from the drop down list, enter the search text and click on "Search" Button. The list of available configurations in the table **mstmobdata** matching the criteria will be displayed in a tabular format. Currently the maximum search output is 20, if the search output results exceeds limit of 20 records, a notification will be displayed in the bottom of the wizard, asking user to further filter the search criteria.



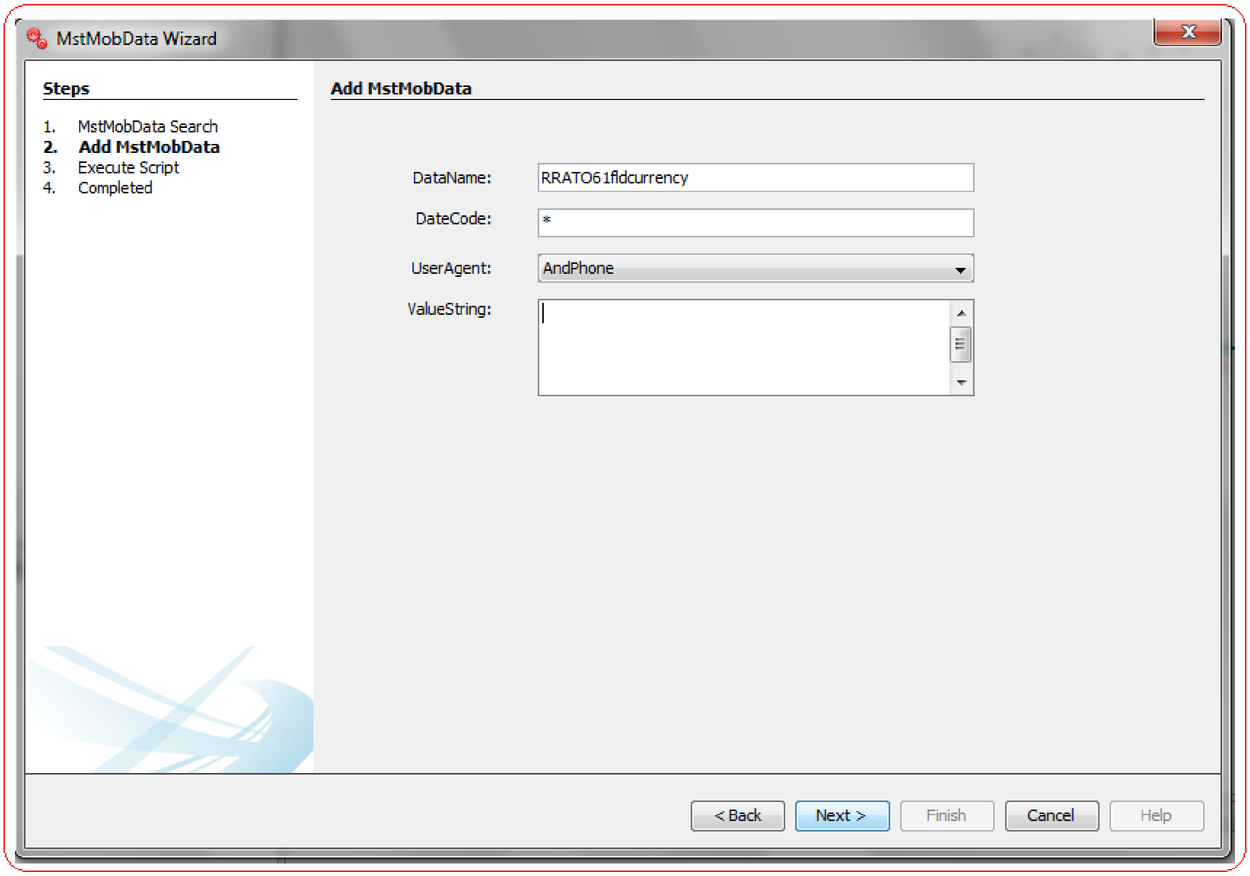
Depending upon the search output one can do the following operations:

I. **Add New:** Select this radio button to add new configuration in database table.
II. **Modify:** This radio button will be enabled only if there is any output to the search action. One can use this action to modify the details of the selected record.
III. **Delete:** This radio button will be enabled only if there is any output to the search action. One can use this action to delete the selected record from the database table.

Click **Next** to go to next step.

MstMobData Registration - Edit

This step is meant to allow users to input details for a new **MstMobData** configuration, modify details of an existing one or view details of an existing one if user wants to delete the same.



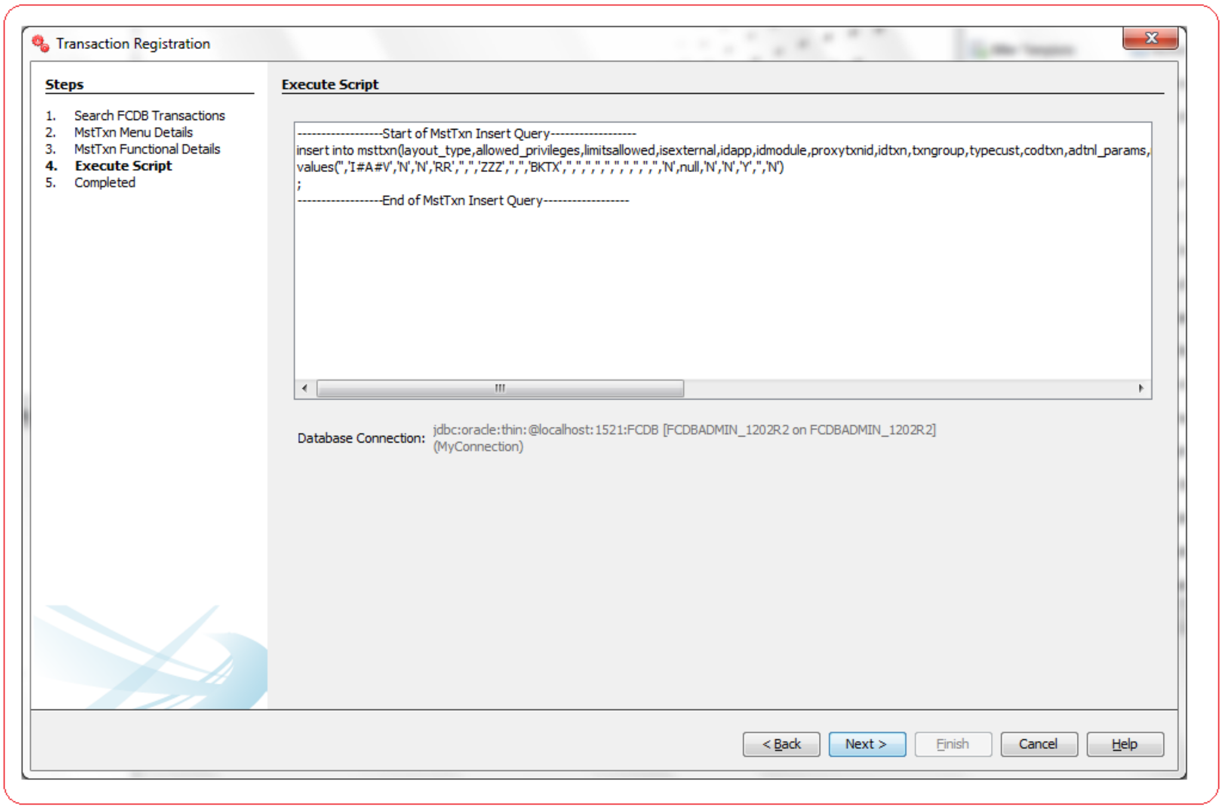
List of the User Input fields and their description are as follows:

- **Data Name**: Enter the Dataname value.
- **Data Code**: Enter the Datacode value.
- **User Agent**: Select the useragent from the dropdown list.
- **Value String**: Enter the value String value.

Click **Next** to go to next step.

Execute Script

This step is meant to display generated script to the user. Generated script is displayed in the text area for the user to verify. Clicking **Next** shall execute the script.



Click **Next** to go to next step.

Confirm Panel

This step is meant to display the location of the file where the scripts are stored and if the script is successfully executed a confirmation message is shown.

